Proceedings of Machine Learning Research vol TBD:1–36, 2020

Pan-Private Uniformity Testing

Kareem Amin KAMIN@GOOGLE.COM
Matthew Joseph MTJOSEPH@GOOGLE.COM
Jieming Mao MAOJM@GOOGLE.COM
Google New York

Abstract

A centrally differentially private algorithm maps raw data to differentially private outputs. In contrast, a locally differentially private algorithm may only access data through public interaction with data holders, and this interaction must be a differentially private function of the data. We study the intermediate model of *pan-privacy*. Unlike a locally private algorithm, a pan-private algorithm receives data in the clear. Unlike a centrally private algorithm, the algorithm receives data one element at a time and must maintain a differentially private internal state while processing this stream.

First, we show that pan-privacy against multiple intrusions on the internal state is equivalent to sequentially interactive local privacy. Next, we contextualize pan-privacy against a single intrusion by analyzing the sample complexity of uniformity testing over domain $[k]$. Focusing on the dependence on k, centrally private uniformity testing has sample complexity $\Theta(\sqrt{k})$, while noninteractive locally private uniformity testing has sample complexity $\Theta(k)$. We show that the sample complexity of pan-private uniformity testing is $\Theta(k^{2/3})$. By a new $\Omega(k)$ lower bound for the sequentially interactive setting, we also separate pan-private from sequentially interactive locally private and multi-intrusion pan-private uniformity testing.

Keywords: Differential privacy, local differential privacy, pan-privacy, uniformity testing

1. Introduction

Differential privacy (Dwork et al. (2006)) promises that a randomized algorithm's output distribution is relatively insensitive to small changes in its input data. This insensitivity hides the presence or absence of individual data elements and provides privacy for the contributors of that data. Rigorous privacy guarantees have driven increasing adoption of differential privacy by industry (Apple (2017); Bittau et al. (2017); Ding et al. (2017); Guevara (2019)), government (Abowd (2018)), and academic researchers (Murtagh et al. (2018); Messing et al. (2019)).

In *central differential privacy* (Dwork et al. (2006)), the algorithm receives a database in the clear, and privacy only constrains the algorithm's eventual output. Central privacy therefore offers the highest utility – for example, the lowest error or sample complexity – but weakest privacy guarantee. In particular, in many real-world applications the input database is acquired over time, and raw data is kept until the time arrives to produce (differentially private) outputs. A user may worry that raw data sitting with a trusted algorithm operator may still be at risk of exfiltration by subpoena, "mission creep" by the operator that contravenes users' original wishes, or a change in operator ownership. Since central privacy makes no guarantees about the intermediate representation of the data during processing, it offers no protection against these events.

One solution to this family of problems is *local differential privacy* (Dwork et al. (2006); Kasiviswanathan et al. (2011)). Locally differentially private algorithms do not receive a database in the clear. Instead, data remains distributed among users, and the algorithm must learn about the data by interacting with these users in a public yet privacy-preserving way. Because users are in charge of randomizing their communications in the protocol, they no longer need to trust an algorithm operator. Unfortunately, this strong privacy guarantee often incurs a significant utility cost. For example, one can compute the sum of n bits to $O(1/\varepsilon)$ additive error under ε-central privacy but, for constant ε, must incur $\Omega(\sqrt{n})$ error under ε-local privacy (Chan et al. (2012)).

We study *pan-privacy* (Dwork et al. (2010b)) as a middle ground in this tradeoff between privacy and utility. A pan-private algorithm receives a stream of raw data (for example, the gradual data acquisition process mentioned above). Pan-privacy has two requirements. First, while processing the data a pan-private algorithm must maintain an internal state that is differentially private against any single intrusion. Second, a pan-private algorithm must ultimately produce a differentially private output.

Central, pan-, and local privacy therefore correspond to different trust models. If a user trusts the algorithm operator to not only perform the computation in question but to responsibly steward raw data in the future, then central privacy is a sufficient guarantee. If a user currently trusts the operator, but also wants to protect themselves against unknown future complications in data stewardship, pan-privacy suffices. For a user who does not trust the operator at all, only local privacy is enough.

1.1. Contributions

We give several results about the relative merits of these models. Taken together, they suggest pan-privacy as a middle ground for both privacy and utility between the central and local models.

1. Through constructive transformations in both directions, we show that pan-privacy against multiple intrusions is equivalent to sequentially interactive local privacy (Section 3).

2. We give matching (in k) upper and lower bounds showing that uniformity testing — the problem of distinguishing uniform and non-uniform distributions through sample access — has pan-private sample complexity $\Theta(k^{2/3})$. The best known locally private uniformity tester achieves $\Theta(k)$ sample complexity by reducing uniformity testing to binary testing (Acharya et al. (2019a)), while the optimal centrally private uniformity tester gets $\Theta(\sqrt{k})$ without reducing the problem domain at all (Acharya et al. (2018)). Our pan-private uniformity tester intermediates between these approaches by reducing uniformity testing over $[k]$ to, roughly, uniformity testing over $[k^{2/3}]$ (Section 4). Our lower bound adapts the approach used by Diakonikolas et al. (2019) to prove testing lower bounds under memory and communication restrictions (Section 5.1).

3. By a new lower bound, again adapting the memory-restricted lower bound of Diakonikolas et al. (2019), we show that sequentially interactive locally private uniformity testing has sample complexity $\Theta(k)$ (Section 5.2).

We briefly elaborate on the first contribution. We view this result as dictating the scope of when pan-privacy is reasonable. If a user requires privacy against multiple intrusions, then the operator suffers no utility loss by using an algorithm that is locally private instead of an algorithm that is pan-private against multiple intrusions. However, there are cases where a user may be satisfied with pan-privacy against a single intrusion. To see why, we use the following simple result.

Fact 1 *Suppose a user's data is element s_t of an (ε, δ)-pan-private algorithm $\mathcal{A}$'s stream. We say an intrusion occurs at time t if the intrusion occurs immediately after $\mathcal{A}$ updates its internal state to i_t after seeing element s_t. If*

1. *the first intrusion (possibly of many) occurs at time $t' \geq t$, or*
2. *all intrusions occur at times $t' < t$,*

then the intruder's view is an (ε, δ)-differentially private function of s_t.

Proof Pan-privacy guarantees that i_t is an (ε, δ)-differentially private function of s_t. In Case 1, the adversary only sees a post-processing of i_t. Differential privacy's resilience to post-processing (see e.g. Proposition 2.1 in Dwork and Roth's survey (Dwork et al. (2014))) implies that this view is (ε, δ)-differentially private in s_t. In Case 2, the adversary's view is independent of s_t, so (ε, δ)-differential privacy is immediate. ■

By Fact 1, if $\mathcal{A}$ is pan-private against a single intrusion, then it guarantees privacy for users who either contribute data before the first intrusion or after all intrusions. However, pan-privacy is not sufficient to protect a user's privacy if the operator has already been compromised and may be compromised again. The key parameter for pan-privacy is therefore the user's trust in the operator when the user contributes their data. This motivates the trust model described in the introduction: if a user trusts the operator today, but wants to "future-proof" themselves for tomorrow, then pan-privacy is a reasonable privacy guarantee.

1.2. Related Work

We start with previous work on pan-privacy. Dwork et al. (2010b) introduced pan-privacy and gave pan-private algorithms for several different counting problems over streams. They also gave two lower bounds. First they separated pan-privacy against one and two intrusions by showing that estimating the number of distinct elements in a stream is much harder with multiple intrusions. Second, they gave a problem, inner product counting, that separates pan-privacy from noninteractive local privacy. Mir et al. (2011) extended these results to new counting problems and dynamic streams. They also showed that pan-private algorithms cannot approximate distinct element count to additive accuracy $o(\sqrt{|X|})$ for data universe X. This improved upon the $\Omega(\sqrt{|X|}/\log(|X|))$ lower bound given by McGregor et al. (2010) for two-party differential privacy (a weaker guarantee than pan-privacy), which was the first separation between central and pan-privacy. Dwork et al. (2010a) also studied pan-privacy, albeit under the additional constraint of continual observation, which requires the algorithm to provide accurate answers after every stream element. They and Chan et al. (2011) gave both upper and lower bounds for counting problems under continual observation.

Our work departs from the above in a few ways. First, we generalize previous results on pan-privacy against two intrusions by showing that it is equivalent to a different model, sequentially interactive local privacy. Second, the testing problems we study focus on learning from samples generated by some distribution, as opposed to previous work on adversarial streaming problems. This distributional quality necessitates different lower bound techniques.

In uniformity testing, a line of work (Goldreich and Ron (2000); Paninski (2008); Valiant and Valiant (2014)) has established that uniformity testing (without privacy) has sample complexity

$\Theta\left(\frac{\sqrt{k}}{\alpha^2}\right)$ where k is the domain size and α is the total variation distance parameter (for more information on testing, see the survey by Canonne (2015)). Acharya et al. (2018) showed that ε-centrally private uniformity testing has sample complexity $\Theta\left(\frac{\sqrt{k}}{\alpha^2} + \frac{\sqrt{k}}{\alpha\sqrt{\varepsilon}} + \frac{k^{1/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{1}{\alpha\varepsilon}\right)$. Acharya et al. (2019a) showed that noninteractive ε-locally private uniformity testing has sample complexity $\Theta\left(\frac{k}{\alpha^2\varepsilon^2}\right)$. Acharya et al. (2019b) proved similar results with special attention to the amount of public randomness. A comparison of our results to this previous work appears in Figure 1.

In the data structures community, several works have studied *history independence* (Micciancio (1997); Naor and Teague (2001); Blelloch and Golovin (2007)). A history independent data structure is one whose memory representation reveals no more information than its abstract representation does. For example, without history independence, the abstract representation of a dictionary may only reveal keys and values while the memory representation also reveals insertion order. Pan-privacy instead aims to guarantee that the abstract representation is a differentially private function of the input data.

Setting	Previous Work	This Work
Non-private	$\Theta\left(\frac{\sqrt{k}}{\alpha^2}\right)$ (Goldreich and Ron (2000); Paninski (2008); Valiant and Valiant (2014))	–
ε-central privacy	$\Theta\left(\frac{\sqrt{k}}{\alpha^2} + \frac{\sqrt{k}}{\alpha\sqrt{\varepsilon}} + \frac{k^{1/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{1}{\alpha\varepsilon}\right)$ (Acharya et al. (2018))	–
ε-pan-privacy	– –	$O\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2} + \frac{\sqrt{k}}{\alpha\varepsilon}\right)$ $\Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2} + \frac{1}{\alpha\varepsilon}\right)$
SI ε-local privacy	$O\left(\frac{k}{\alpha^2\varepsilon^2}\right)$ (Acharya et al. (2019a))	$\Omega\left(\frac{k}{\alpha^2\varepsilon^2}\right)$
NI ε-local privacy	$\Theta\left(\frac{k}{\alpha^2\varepsilon^2}\right)$ (Acharya et al. (2019a))	–

Figure 1: A comparison of the uniformity testing sample complexity bounds given in this and previous work. "SI" is sequentially interactive and "NI" is noninteractive. Before this work, no pan-private bounds were known, and it was not known that $O\left(\frac{k}{\alpha^2\varepsilon^2}\right)$ is tight for sequentially interactive protocols.

2. Preliminaries

2.1. Central Differential Privacy

A randomized algorithm $\mathcal{A}$ satisfies central differential privacy if it maps raw databases to outcomes such that the distribution over outcomes is relatively insensitive to small changes in the database.

This insensitivity, which hides the presence or absence of any one user, provides the privacy guarantee.

Definition 1 (Central differential privacy (Dwork et al. (2006))) *Given data universe $\mathcal{X}$ and two databases $D, D' \in \mathcal{X}^n$, D and D' are* neighbors *if they differ in ≤ 1 element. Given algorithm $\mathcal{A} \colon \mathcal{X}^n \to Y$, $\mathcal{A}$ is (ε, δ)*-differentially private *if for all subsets $S \subset Y$,*

$$\mathbb{P}_{\mathcal{A}}\left[\mathcal{A}(D) \in S\right] \leq e^{\varepsilon}\mathbb{P}_{\mathcal{A}}\left[\mathcal{A}(D') \in S\right] + \delta.$$

For this work, it is important to note that centrally private algorithms enjoy trusted (central) access to the entire raw database. In particular, they may perform arbitrary computations on raw data before releasing a private output. Pan- and locally private algorithms have restricted forms of access to the data.

2.2. Pan-privacy

A *pan-private* algorithm operates in a different setting with different guarantees. Here, the algorithm $\mathcal{A}$ receives the database as a stream, one element at a time, and updates its internal state after seeing each element. The element is then deleted from $\mathcal{A}$'s memory, and $\mathcal{A}$ continues processing the stream[1]. The entirety of $\mathcal{A}$'s knowledge about the stream so far is thus contained in this internal state. At the end of the stream, $\mathcal{A}$ produces an output as its final answer. Pan-privacy mandates that $\mathcal{A}$'s internal state and final answer must be differentially private functions of the stream on a per-element basis.

Definition 2 (Pan-privacy (Dwork et al. (2010b))) *Let $\mathcal{X}$ be a data universe, and let $\mathcal{S} = \mathcal{X}^{\mathbb{N}}$ be the set of streams from $\mathcal{X}$. Two streams $s, s' \in \mathcal{S}$ are* neighbors *if there exist x and $x' \in \mathcal{X}$ such that replacing a single instance of $x \in s$ with x' produces s'.*

A pan-private algorithm consists of an internal algorithm $\mathcal{A}_{\mathcal{I}}$ and an output algorithm $\mathcal{A}_{\mathcal{O}}$. $\mathcal{A}$ maps streams to internal states by repeated application of $\mathcal{A}_{\mathcal{I}}$, which maps an internal state and element of $\mathcal{X}$ to an internal state, $\mathcal{A}_{\mathcal{I}} \colon \mathcal{I} \times \mathcal{X} \to \mathcal{I}$. At some time the stream ends and $\mathcal{A}$ publishes a final output $\mathcal{A}_{\mathcal{O}}(i)$ where i is the internal state of $\mathcal{A}$ at the end of the stream. For stream s, let $\mathcal{A}_{\mathcal{I}}(s)$ denote the internal state of $\mathcal{A}$ after processing s, and let $s_{\leq t}$ denote the first t elements of stream s. $\mathcal{A}$ is (ε, δ)-pan-private *if, for any neighboring streams s and s', any time t, and any set $E \subset \mathcal{I} \times \mathcal{O}$*

$$\mathbb{P}_{\mathcal{A}}\left[(\mathcal{A}_{\mathcal{I}}(s_{\leq t}), \mathcal{A}_{\mathcal{O}}(\mathcal{A}_{\mathcal{I}}(s))) \in E\right] \leq e^{\varepsilon}\mathbb{P}_{\mathcal{A}}\left[(\mathcal{A}_{\mathcal{I}}(s'_{\leq t}), \mathcal{A}_{\mathcal{O}}(\mathcal{A}_{\mathcal{I}}(s'))) \in E\right] + \delta. \tag{1}$$

This paper will focus on pure pan-privacy, where $\delta = 0$. We shorthand this as ε-pan-privacy.

Pan-privacy thus protects against an adversary that sees any single internal state of $\mathcal{A}$ as well as its final output. The second requirement implies that any pan-private algorithm is also centrally private; the key additional contribution of pan-privacy is the maintenance of the differentially private internal state. To generalize Definition 2 to $c > 1$ intrusions, we can replace inequality 1 with

$$\mathbb{P}_{\mathcal{A}}\left[(\mathcal{A}_{\mathcal{I}}(s_t)_{t=t_1}^{t_c}, \mathcal{A}_{\mathcal{O}}(\mathcal{A}_{\mathcal{I}}(s))) \in E\right] \leq e^{\varepsilon}\mathbb{P}_{\mathcal{A}}\left[(\mathcal{A}_{\mathcal{I}}(s'_t)_{t=t_1}^{t_c}, \mathcal{A}_{\mathcal{O}}(\mathcal{A}_{\mathcal{I}}(s'))) \in E\right] + \delta$$

[1] As is standard in pan-privacy, we assume that the process of receiving an element and updating the state is atomic: the adversary cannot intrude on the internal state between the reception of a new stream element and the internal state update. Without this assumption, nothing prevents the adversary from possibly seeing a data point in the clear, and differential privacy is impossible.

where $E \subset \mathcal{I}^c \times \mathcal{O}$.

We note that our definition of pan-privacy differs from the original. This stems from slightly different goals. The original work of Dwork et al. (2010b) focused on tracking statistics of a stream of unknown length and allowed for the possibility that the stream could end unexpectedly. They also allowed for multiple outputs by the algorithm. We instead analyze problems from a sample complexity perspective and focus on the number of samples needed to solve a problem pan-privately. This leads us to consider streams of fixed length (determined by the algorithm as the required sample complexity) and a single output (the answer to the problem in question). Additionally, Dwork et al. (2010b) studied streams where each element is a (user, value) pair and a neighboring stream may replace all values contributed by any one user. We remove the notion of a user and simply view a stream as a sequence of elements. We therefore guarantee element-level rather than user-level privacy[2]. Nonetheless, the basic idea of pan-privacy – its privacy against an adversary who sees a single internal state and the output – remains intact.

2.3. Local Differential Privacy

A *locally differentially private* algorithm satisfies a still more restrictive privacy guarantee. Unlike pan-private algorithms, locally private algorithms never see any data in the clear. Instead, a locally private algorithm is a public interaction between users, each of whom privately holds a single data element. Since our main point of comparison is pan-private algorithms, we view these users as stream elements. A pan-private algorithm sees each stream element, but a locally private algorithm only sees the *randomizer* output produced by each stream element. However, this is a difference only in presentation, and a user obtains the same kind of local privacy guarantee whether we view them as a user or a stream element. Up to this difference, our local differential privacy definitions generally imitate those given by Joseph et al. (2019).

Definition 3 *An* (ε, δ)-randomizer $R\colon X \to Y$ *is an* (ε, δ)*-differentially private function taking a single data point as input.*

Because communication occurs only through randomizers, the overall record of public interaction is private. We more formally study this interaction in terms of its *transcript*.

Definition 4 *A* transcript π *is a vector of tuples* (R_t, y_t) *indicating the randomizer used and output produced at each time* t*.*

We can then view a locally private protocol as a coordinating mechanism that takes a transcript and selects a randomizer for the next stream element.

Definition 5 *Let* S_π *denote the collection of transcripts and* S_R *the collection of randomizers. Then a* protocol $\mathcal{A}$ *is a function* $\mathcal{A}\colon S_\pi \to S_R$ *mapping transcripts to randomizers.*

A locally private protocol generally includes some post-processing of the transcript to generate some final output. Since this post-processing is still a function of the transcript, we abstract it away and focus only on the transcript. Next, we distinguish between different notions of interactivity for locally private protocols.

[2] Note that this allows the closest comparison with existing centrally and locally private uniformity testers, which all employ element-level privacy.

Definition 6 *If locally private protocol $\mathcal{A}$ makes all randomizer assignments before the stream begins (i.e., each randomizer choice R_t is independent of the transcript so far conditioned on t) then $\mathcal{A}$ is* noninteractive. *If $\mathcal{A}$ makes these assignments adaptively as the stream progresses, then $\mathcal{A}$ is* sequentially interactive *(Duchi et al. (2013)).*

The most general model of local privacy allows *full interactivity*: users may produce arbitrarily many outputs in arbitrary sequences. In particular, a protocol may re-query past participants. This is analogous to processing a stream with multiple passes. Since we focus on pan-privacy in the single-pass model, we will compare it to noninteractive and sequentially interactive locally private protocols, which can only query each participant at most once. We now formally define local differential privacy.

Definition 7 *A protocol $\mathcal{A}$ is* (ε, δ)-locally differentially private *if its transcript is an (ε, δ)-differentially private function of the user data. If $\delta = 0$, we say $\mathcal{A}$ is* ε-locally differentially private.

In particular, a sequentially interactive protocol is (ε, δ)-locally differentially private if and only if each randomizer used is an (ε, δ)-randomizer.

3. Pan-privacy and Local Privacy

We first show that any algorithm that is pan-private against multiple intrusions has a *locally* private equivalent (Theorem 8). The main idea is that the operator of a pan-private algorithm $\mathcal{A}_{2P}$ cannot know when two intrusions will occur. In particular, if the two intrusions occur at times t and $t+1$ — respectively, immediately after $\mathcal{A}_{2P}$ processes s_t and s_{t+1} — then failure to randomize the internal state between t and $t+1$ may reveal element s_{t+1}. The operator must therefore re-randomize the state at *every* time step.

We briefly sketch the proof of Theorem 8 (full proofs of this and other results appear in the Appendix). First, we observe that any $\mathcal{A}_{2P}$ that is ε-pan-private against two intrusions can be modified into an algorithm $\mathcal{A}_{1P}$ that maintains all of its internal states thus far and still remain ε-pan-private against *one* intrusion (Lemma 16). Because this single intrusion may come at the end of the stream, the complete list of internal states during the stream must be an ε-differentially private function of the stream. We can therefore simulate this procedure in the sequentially interactive local model and have the transcript generate this complete list of internal states (Lemma 17).

In the other direction, we convert any ε-sequentially interactive locally private protocol $\mathcal{A}_L$ to $\mathcal{A}_{2P}$, which is ε-pan-private against two intrusions. $\mathcal{A}_{2P}$ simulates $\mathcal{A}_L$ and stores the transcript so far as its internal state. Since this transcript is an ε-differentially private function of the data (recall that the transcript for $\mathcal{A}_L$ is public), $\mathcal{A}_{2P}$ is ε-pan-private against an arbitrary number of intrusions onto its internal state.

Theorem 8 *For every $\mathcal{A}_{2P}$ that is ε-pan-private against two intrusions and generates output distribution O given input stream s, there exists $\mathcal{A}_L$ that is sequentially interactive ε-locally private and generates transcript distribution O given s, and vice-versa.*

Proof See Appendix, Section 7. ■

4. Uniformity Testing

We now turn to upper bounds for pan-privacy against a single intrusion. Our benchmark problem is *uniformity testing*. In uniformity testing, a tester receives i.i.d. sample access to an unknown discrete distribution p over $[k]$ and must determine with nontrivial constant probability whether p is uniform or α-far from uniform in total variation distance. Below, let U_k denote the uniform distribution over $[k]$.

Definition 9 (Uniformity testing) *An algorithm $\mathcal{A}$ is a* uniformity tester on m samples *if, given m i.i.d. samples from p,*

1. *when $p = U_k$, with probability $\geq 2/3$ $\mathcal{A}$ outputs "uniform", and*
2. *when $||p - U_k||_{TV} \geq \alpha$, with probability $\geq 2/3$ $\mathcal{A}$ outputs "non-uniform".*

The specific choice of $2/3$ is arbitrary. The important point is that there is a constant separation between output probabilities, which can be amplified to $2/3$ with a constant number of repetitions. We therefore focus on achieving any such constant separation. Details for this standard perspective appear in Appendix 8.

4.1. Warmup: SIMPLEPANTEST

We start with a suboptimal uniformity tester SIMPLEPANTEST. SIMPLEPANTEST is a warmup and eventual building block for a better algorithm PANTEST (Section 4.2).

Like many uniformity testers, SIMPLEPANTEST computes a statistic on the data and compares it to a threshold. The statistic is designed to be small when p is uniform and large if p is α-far from uniform. For SIMPLEPANTEST, our statistic is

$$Z' = \sum_{i=1}^{k} \frac{(H_i - m/k)^2 - H_i}{m/k}$$

where m is the number of samples and H is a noisy histogram over $[k]$ where bin i counts the number of occurrences of element i in the stream. H contains Laplace noise added to each bin both before and after the stream. The first addition of noise ensures the privacy of the internal states during the stream, while the second addition of noise is for the privacy of the final output. Pseudocode for SIMPLEPANTEST appears below; values for m and T_U are determined in the proof of Lemma 10.

Inspired by similar statistics in non-private testing (Acharya et al. (2013); Chan et al. (2014); Acharya et al. (2015)), Cai et al. (2017) originally studied Z' for centrally private identity testing. However, they lower bounded its variance and argued that high variance makes it a suboptimal centrally private tester. We instead upper bound its variance and show that Z' yields a nontrivial pan-private uniformity tester.

Our argument is simple. First, we upper bound the variance of Z'. We then apply Chebyshev's inequality to upper bound Z' when p is uniform and lower bound Z' when p is α-far from uniform. These bounds drive our choice of the threshold T_U. We then compute the number of samples m required to separate these quantities on either side of T_U.

Finally, note that we actually draw $m' \sim \text{Poisson}\,(m)$ samples, not m. This "Poissonization" trick is important for the analysis used to prove Lemma 19. Since Poisson (m) concentrates around

Algorithm 1 Pan-private uniformity tester SIMPLEPANTEST

```
Require: privacy parameter $\varepsilon$, domain $[k]$
    Set sample size $m' \sim \mathsf{Poisson}\,(m)$ and threshold $T_U$
    Initialize private histogram $H \leftarrow \mathsf{Lap}\left(\frac{1}{\varepsilon}\right)^k \in \mathbb{R}^k$
    for stream elements $s_t = s_1, \dots, s_{m'}$ do
        $H_{s_t} \leftarrow H_{s_t} + 1$
    end for
    $H \leftarrow H + \mathsf{Lap}\left(\frac{1}{\varepsilon}\right)^k \in \mathbb{R}^k$
    $Z' \leftarrow \sum_{i=1}^k \frac{(H_i - m/k)^2 - H_i}{m/k}$
    if $Z' > T_U$ then
        Output "non-uniform"
    else
        Output "uniform"
    end if
```

m (Canonne (2016)), a uniformity tester on $\mathsf{Poisson}\,(m)$ samples implies a uniformity tester on a constant factor more samples with a constant decrease in success probability (see Section D.4 in the survey of Canonne (2015) for a more detailed discussion of Poissonization).

Lemma 10 *For $m = \Omega\left(\frac{k^{3/4}}{\alpha\varepsilon} + \frac{\sqrt{k}}{\alpha^2}\right)$,* SIMPLEPANTEST *is an ε-pan-private uniformity tester on m samples.*

Proof See Appendix, Section 9. ■

4.2. Optimal pan-private tester: PANTEST

We now use SIMPLEPANTEST as a building block for a more complex tester PANTEST. At a high level, PANTEST splits the difference between local and central uniformity testers. We briefly recap these approaches for context.

Centrally private uniformity testers compute a fine-grained statistic depending on the empirical counts of each element $i \in [k]$. Specific methods include χ^2-style statistics (Cai et al. (2017)), collision-counting (Aliakbarpour et al. (2018)), and empirical total variation distance from U_k (Acharya et al. (2018)), but all of these methods depend on accurate counts for each $i \in [k]$. Cai et al. (2017) observed that adding Laplace noise to each such count before analyzing the statistic is centrally private. The cost is a large decrease in accuracy. This is unfortunate in our pan-private setting, as pan-privacy appears to force the same kind of per-count noise. Intuitively, a pan-private tester might benefit by maintaining a coarser statistic — i.e., one that tracks fewer counts — that is easier to maintain privately.

The best known[3] locally private uniformity tester, due to Acharya et al. (2019a), uses an extreme version of this coarser strategy. Their approach randomly halves the domain $[k]$ into sets U and U^c and compares the number of samples falling into each. They prove that if p is sufficiently non-uniform to start, then $p(U)$ and $p(U^c)$ will also be non-uniform — albeit to a much smaller degree

[3] Note that existing lower bounds, including those in this paper, have not ruled out the possibility that a *fully interactive* locally private uniformity tester obtains better sample complexity.

— with constant probability. This reduces uniformity testing to a simpler binary testing problem that, because of its much smaller domain, is more amenable to local privacy. However, it does so at the cost of a large reduction in testing distance, which makes the core distinguishing problem harder. Thus both locally private and pan-private versions of this approach have sample complexity $\Omega(k)$. Intuitively, because pan-privacy does not force as much noise as local privacy, a pan-private algorithm might benefit by maintaining a finer statistic.

PANTEST capitalizes on both of these ideas. First, it randomly partitions $[k]$ into n groups $G_1, \dots, G_n$ of size $\Theta(k/n)$. It then runs SIMPLEPANTEST to test uniformity of the induced distribution over $[n]$, treating samples falling in each G_j as samples of $j \in [n]$.

PANTEST thus intermediates between the central and local approaches. It chooses $n = n(\alpha, \varepsilon, k)$ according to $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}}$. When $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} < 2$, $n(\alpha, \varepsilon, k) = 2$ and PANTEST uses the half-partition approach from local privacy. When $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} > k$, then $n(\alpha, \varepsilon, k) = k$ and PANTEST uses the unpartitioned approach from central privacy. Finally, when $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} \in [2, k]$, $n(\alpha, \varepsilon, k) = \lfloor \frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} \rfloor$ and PANTEST takes a middle ground. These choices enable PANTEST to calibrate the noise contributed by privately maintaining different counts with the testing distance α Making this tradeoff work relies crucially on the $O\left(\frac{1}{\alpha}\right)$ dependence on distance achieved by SIMPLEPANTEST in its $k^{3/4}$ term. In contrast, the $\Omega\left(\frac{k}{\alpha^2}\right)$ dependence of the best known locally private uniformity tester yields no improvement with this approach. Pseudocode for PANTEST appears below.

Algorithm 2 Improved pan-private uniformity tester PANTEST

```
Require: privacy parameter $\varepsilon$, domain $[k]$
  if $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} < 2$ then
      $n \leftarrow 2$
  else if $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} > k$ then
      $n \leftarrow k$
  else
      $n \leftarrow \lfloor \frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} \rfloor$
  end if
  Randomly partition $[k]$ into $n$ groups $G_1, \dots, G_n$ of size $\Theta(k/n)$
  Run SIMPLEPANTEST($\varepsilon$, $[n]$), treating each element $s_t \in G_j$ as $j \in [n]$
```

For this reduction to work, the aforementioned decrease in testing distance between $[k]$ and $[n]$ must not be too large. We show this in Lemma 11, which generalizes a similar result of Acharya et al. (2019a) for the special case of a partition into two subsets. As pointed out by a reviewer, this generalization is not new (see Theorem 3.2 from Acharya et al. (2019b)), but we include our proof here for completeness.

Lemma 11 *Let p be a distribution over $[k]$ such that $||p - U_k||_{TV} = \alpha$ and let $G_1, \dots, G_n$ be a uniformly random partition of $[k]$ into $n > 1$ subsets of size $\Theta(k/n)$. Define induced distribution p_n over $[n]$ by $p_n(j) = \sum_{i \in G_j} p(i)$ for each $j \in [n]$. Then, with probability $\geq \frac{1}{954}$ over the selection of $G_1, \dots, G_n$,*

$$||p_n - U_n||_{TV} = \Omega\left(\alpha\sqrt{\frac{n}{k}}\right).$$

Proof See Appendix, Section 9. ■

Due to the $1/954$ success probability of Lemma 11, we have a smaller (but still constant) separation between output probabilities. We thus use the amplification argument discussed after Definition 9 to get Theorem 12. The guarantee combines Lemma 11 with Lemma 10, substituting n for k and $\alpha\sqrt{\frac{n}{k}}$ for α.

Theorem 12 *For $m = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2} + \frac{\sqrt{k}}{\alpha\varepsilon}\right)$,* PANTEST *is an ε-pan-private uniformity tester on m samples.*

Proof See Appendix, Section 9. ■

5. Lower Bounds

We now turn to lower bounds. Our first result gives a tight (in k) $\Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}}\right)$ lower bound for ε-pan-private testing (Section 5.1). Our second result extends the previous $\Omega\left(\frac{k}{\alpha^2\varepsilon^2}\right)$ lower bound for noninteractive (ε, δ)-locally private uniformity testing (Acharya et al. (2019a)) to the sequentially interactive case (Section 5.2).

Both of our lower bounds adapt the approach used by Diakonikolas et al. (2019) to prove testing lower bounds under memory restrictions and communication restrictions. Like Diakonikolas et al. (2019), we consider the problem of distinguishing between two distributions. If uniform random variable X is 0 then the distribution is uniform. If X is 1 then each element has probability mass slightly perturbed from uniform such that the distribution is α-far from uniform in total variation distance. Our argument then proceeds by upper bounding the mutual information between the random variable X and the algorithm's internal state (in the pan-private case) or transcript (in the locally private case). Controlling this quantity lower bounds the number of samples required to identify X. This gives the final uniformity testing sample complexity lower bounds.

The main difference in our lower bounds is that Diakonikolas et al. (2019) restrict their algorithm to use an internal state with b bits of memory. This memory restriction immediately implies that the internal state's entropy (and thus its mutual information with any other random variable) is also bounded by b. In our case, we must use our privacy restrictions to replace this result. Doing so constitutes the bulk of our arguments.

Finally, we note that these results add to lines of work conceptually connecting restricted memory to pan-privacy (Dwork et al. (2010b); Mir et al. (2011)) and connecting restricted communication to local privacy (McGregor et al. (2010); Acharya et al. (2019a); Duchi and Rogers (2019); Acharya et al. (2019b); Joseph et al. (2020)).

5.1. Pan-private Lower Bound

We start with the pan-private lower bound. While we state our result using $\alpha \leq 1/2$, the choice of $1/2$ is arbitrary: the same argument works for any α bounded below 1 by a constant. A short primer on the information theory used in our argument appears in Appendix 11.

Theorem 13 *For $\varepsilon = O(1)$ and $\alpha \leq 1/2$, any ε-pan-private uniformity tester requires $m = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2} + \frac{1}{\alpha\varepsilon}\right)$ samples.*

Proof See Appendix, Section 10. ■

5.2. Locally Private Lower Bound

We now move to the locally private lower bound. We state our result for ε-locally private algorithms, but this is without loss of generality by the work of Bun et al. (2018) and Cheu et al. (2019), which demonstrates an equivalence between (ε, δ)- and $(\varepsilon, 0)$-local privacy for reasonable parameter ranges. A concise statement of these results also appears as Lemma 5.2 in the work of Joseph et al. (2019).

At a high level, the main difference the pan-private and sequentially interactive lower bounds is that the locally private algorithm does not see any sample S_t. Instead, the algorithm sees a randomizer output based on S_t. We can therefore use past work quantifying the information loss between a randomizer's input and output (Duchi et al. (2013)) to bound information learned more tightly than under pan-privacy. This partially explains, for example, the locally private lower bound's different dependence on ε. Replacing the memory upper bound used by Diakonikolas et al. (2019) with the local privacy restriction also requires a different argument than in the pan-private case.

Theorem 14 *For $\varepsilon = O(1)$, any sequentially interactive ε-locally private uniformity tester requires $m = \Omega\left(\frac{k}{\alpha^2\varepsilon^2}\right)$ samples.*

Proof See Appendix, Section 10. ■

6. Acknowledgments

We thank the reviewers for their helpful comments. These improved the presentation, statements, and proofs of several of our results.

References

John Abowd. Disclosure avoidance for block level data and protection of confidentiality in public tabulations. census.gov/cac/sac/meetings/2018-12/abowd-disclosure-avoidance.pdf, 2018. Accessed: 04/25/2020.

Jayadev Acharya, Ashkan Jafarpour, Alon Orlitsky, and Ananda Suresh. A competitive test for uniformity of monotone distributions. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2013.

Jayadev Acharya, Constantinos Daskalakis, and Gautam Kamath. Optimal testing for properties of distributions. In *Conference on Neural Information Processing Systems (NIPS)*, 2015.

Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Differentially private testing of identity and closeness of discrete distributions. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Jayadev Acharya, Clément Canonne, Cody Freitag, and Himanshu Tyagi. Test without trust: Optimal locally private distribution testing. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019a.

Jayadev Acharya, Clément L Canonne, Yanjun Han, Ziteng Sun, and Himanshu Tyagi. Domain compression and its application to randomness-optimal distributed goodness-of-fit. *arXiv preprint arXiv:1907.08743*, 2019b.

Maryam Aliakbarpour, Ilias Diakonikolas, and Ronitt Rubinfeld. Differentially private identity and equivalence testing of discrete distributions. In *International Conference on Machine Learning (ICML)*, 2018.

Differential Privacy Team Apple. Learning with privacy at scale. Technical report, Apple, 2017.

Raef Bassily and Adam Smith. Local, private, efficient protocols for succinct histograms. In *Symposium on the Theory of Computing (STOC)*, 2015.

Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *Symposium on Operating Systems Principles (SOSP)*, 2017.

Guy E. Blelloch and Daniel Golovin. Strongly history-independent hashing with applications. In *Symposum on the Foundations of Computer Science (FOCS)*, 2007.

Mark Bun, Jelani Nelson, and Uri Stemmer. Heavy hitters and the structure of local privacy. In *Symposium on Principles of Database Systems (PODS)*, 2018.

Bryan Cai, Constantinos Daskalakis, and Gautam Kamath. Priv'it: private and sample efficient identity testing. In *International Conference on Machine Learning (ICML)*, 2017.

Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? In *Electronic Colloquium on Computational Complexity (ECCC)*, 2015.

Clément L Canonne. A short note on poisson tail bounds. http://www.cs.columbia.edu/~ccanonne/files/misc/2017-poissonconcentration.pdf, 2016.

Siu-On Chan, Ilias Diakonikolas, Paul Valiant, and Gregory Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Symposium on Discrete Algorithms (SODA)*, 2014.

T.-H. Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. *ACM Trans. Inf. Syst. Secur.*, 2011.

TH Hubert Chan, Elaine Shi, and Dawn Song. Optimal lower bound for differentially private multi-party aggregation. In *European Symposium on Algorithms (ESA)*, 2012.

Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques (CRYPTO)*, 2019.

Ilias Diakonikolas, Themis Gouleakis, Daniel M. Kane, and Sankeerth Rao. Communication and memory efficient testing of discrete distributions. In *Conference on Learning Theory (COLT)*, 2019.

Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *Conference on Neural Information Processing Systems (NIPS)*, pages 3574–3583, 2017.

John Duchi and Ryan Rogers. Lower bounds for locally private estimation via communication complexity. In *Conference on Learning Theory (COLT)*, 2019.

John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Local privacy and statistical minimax rates. In *Symposium on the Foundations of Computer Science (FOCS)*. IEEE, 2013.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference (TCC)*, 2006.

Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N Rothblum. Differential privacy under continual observation. In *Symposium on the Theory of Computing (STOC)*, 2010a.

Cynthia Dwork, Moni Naor, Toniann Pitassi, Guy N Rothblum, and Sergey Yekhanin. Pan-private streaming algorithms. In *Innovations in Computer Science (ICS)*, 2010b.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 2014.

Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, 2000.

Miguel Guevara. Enabling developers and organizations to use differential privacy. developers.googleblog.com/2019/09/enabling-developers-and-organizations.html, 2019. Accessed: 09-12-2019.

Matthew Joseph, Jieming Mao, Seth Neel, and Aaron Roth. The role of interactivity in local differential privacy. In *Symposium on the Foundations of Computer Science (FOCS)*. IEEE, 2019.

Matthew Joseph, Jieming Mao, and Aaron Roth. Exponential separations in local differential privacy. In *Symposium on Discrete Algorithms (SODA)*, 2020.

Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 2011.

Andrew McGregor, Ilya Mironov, Toniann Pitassi, Omer Reingold, Kunal Talwar, and Salil Vadhan. The limits of two-party differential privacy. In *Symposium on the Foundations of Computer Science (FOCS)*, 2010.

Solomon Messing, Christina DeGregorio, Bennett Hillenbrand, Gary King, Saurav Mahanti, Chaya Nayak, Nathaniel Persily, Bogdan State, and Arjun Wilkins. Facebook privacy-protected urls light table release. socialscience.one/files/partnershipone/files/facebook_urls-light_codebook_v2.0.pdf, 2019. Accessed: 09-18-2019.

Daniele Micciancio. Oblivious data structures: Applications to cryptography. In *Symposium on the Theory of Computing (STOC)*, 1997.

Darakhshan Mir, Shan Muthukrishnan, Aleksandar Nikolov, and Rebecca N Wright. Pan-private algorithms via statistics on sketches. In *Symposium on Principles of Database Systems (PODS)*, 2011.

Jack Murtagh, Kathryn Taylor, George Kellaris, and Salil Vadhan. Usable differential privacy: A case study with psi. *arXiv preprint arXiv:1809.04103*, 2018.

Moni Naor and Vanessa Teague. Anti-persistence: History independent data structures. In *Symposium on the Theory of Computing (STOC)*, 2001.

Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 2008.

Gregory Valiant and Paul Valiant. An automatic inequality prover and instance optimal identity testing. In *Foundations of Computer Science (FOCS)*, 2014.

7. Pan-Local Proof

Theorem 15 *For every $\mathcal{A}_{2P}$ that is ε-pan-private against two intrusions and generates output distribution O given input stream s, there exists $\mathcal{A}_L$ that is sequentially interactive ε-locally private and generates transcript distribution O given s, and vice-versa.*

Proof $\underline{\Rightarrow \text{(pan to local)}}$: We start by converting from pan-privacy against two intrusions to pan-privacy against one intrusion while preserving all internal states.

Lemma 16 *Suppose $\mathcal{A}_{2P}$ is ε-pan-private against two intrusions, and let $I_{2,t}$ be the random variable for the internal state of $\mathcal{A}_{2P}$ after stream element t. Then there exists $\mathcal{A}_{1P}$ that is ε-pan-private against one intrusion such that, for analogously defined $I_{1,t}$, for any stream $s_{\le t}$, the concatenation $I_{2,1} \circ I_{2,2} \cdots \circ I_{2,t}$ is distributed identically to $I_{1,t}$.*

Proof We first define $\mathcal{A}_{1P}$. For $j \in \{1, 2\}$, define $i_{j,t}$ to be the realized internal state of $\mathcal{A}_{jP}$ after seeing the t^{th} stream element. Each internal state $i_{1,t}$ of $\mathcal{A}_{1P}$ is a concatenation of internal states $i_{2,1} \circ \cdots \circ i_{2,t}$, and for any internal state i of $\mathcal{A}_{1P}$ we let i^{-1} denote the most recently concatenated state. For example, for $i = i_{2,1} \circ \cdots \circ i_{2,t}$, $i^{-1} = i_{2,t}$[4]. We then define the internal algorithm of $\mathcal{A}_{1P}$ by $\mathcal{A}_{1P,\mathcal{I}}(i, x) = i \circ \mathcal{A}_{2P,\mathcal{I}}(i^{-1}, x)$. Finally, we define the output algorithm of $\mathcal{A}_{1P}$ by $\mathcal{A}_{1P,\mathcal{O}}(i) = \mathcal{A}_{2P,\mathcal{O}}(i^{-1})$. As a result, $\mathcal{A}_{1P,\mathcal{O}}(\mathcal{A}_{1P,\mathcal{I}}(s)) = \mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s))$, and $\mathcal{A}_{1P}$ and $\mathcal{A}_{2P}$ have identical output distributions.

We will prove this result for discrete state spaces; a similar approach works for continuous state spaces if we replace probability mass functions with densities. To prove ε-pan-privacy of $\mathcal{A}_{1P}$ against one intrusion, it suffices to fix neighboring streams s and s', internal state set i, output state set o, stream position t, and show

$$\frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s_{\le t}) = i\right] \mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{O}}(\mathcal{A}_{1P,\mathcal{I}}(s)) = o \mid \mathcal{A}_{1P,\mathcal{I}}(s_{\le t}) = i\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s'_{\le t}) = i\right] \mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{O}}(\mathcal{A}_{1P,\mathcal{I}}(s')) = o \mid \mathcal{A}_{1P,\mathcal{I}}(s'_{\le t}) = i\right]} \le e^{\varepsilon}.$$

First, by the definition of $\mathcal{A}_{1P}$, it suffices to show

$$\frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s_{\le t}) = i\right] \mathbb{P}_{\mathcal{A}_{2P}}\left[\mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s)) = o \mid \mathcal{A}_{2P,\mathcal{I}}(s_{\le t}) = i^{-1}\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s'_{\le t}) = i\right] \mathbb{P}_{\mathcal{A}_{2P}}\left[\mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s')) = o \mid \mathcal{A}_{2P,\mathcal{I}}(s'_{\le t}) = i^{-1}\right]} \le e^{\varepsilon}. \tag{2}$$

[4] We assume that it is possible to separate a concatenation into states of $\mathcal{A}_{2P}$ after the fact. This assumption is easily (but less neatly) removed using a separator character $\perp$.

Suppose streams s and s' differ at time t^*, i.e. $s_{t^*} \neq s'_{t^*}$. If $t^* > t$, then we immediately have $\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s_{\leq t}) = i\right] = \mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s'_{\leq t}) = i\right]$, and $\frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[\mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s))=o|\mathcal{A}_{2P,\mathcal{I}}(s_{\leq t})=i^{-1}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[\mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s'))=o|\mathcal{A}_{2P,\mathcal{I}}(s'_{\leq t})=i^{-1}\right]} \leq e^{\varepsilon}$ follows from the ε-pan-privacy of $\mathcal{A}_{2P}$. Thus Inequality 2 holds.

The remaining case is when $t^* \leq t$. Here, $\frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[\mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s))=o|\mathcal{A}_{2P,\mathcal{I}}(s_{\leq t})=i^{-1}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[\mathcal{A}_{2P,\mathcal{O}}(\mathcal{A}_{2P,\mathcal{I}}(s'))=o|\mathcal{A}_{2P,\mathcal{I}}(s'_{\leq t})=i^{-1}\right]} = 1$, and we need to upper bound $\frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s_{\leq t})=i\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s'_{\leq t})=i\right]}$. Since $\mathcal{A}_{1P,\mathcal{I}}(s_{\leq t})$ is conditionally independent of $\mathcal{A}_{1P,\mathcal{I}}(s_{\leq t^*-1})$ given $\mathcal{A}_{1P,\mathcal{I}}(s_{\leq t^*})$, it suffices to show that $\frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s_{\leq t^*})=i\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[\mathcal{A}_{1P,\mathcal{I}}(s'_{\leq t^*})=i\right]} \leq e^{\varepsilon}$. Recall that $I_{j,t}$ is the random variable for the internal state of $\mathcal{A}_j$ after seeing the t^{th} stream element. Then it is equivalent to show $\frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[I_{1,t^*}=i|S_{\leq t*}=s_{\leq t*}\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[I_{1,t^*}=i|S_{\leq t^*}=s'_{\leq t*}\right]} \leq e^{\varepsilon}$.

We introduce some additional notation to prove this claim. i is an internal state for $\mathcal{A}_{1P}$ and is therefore a concatenation of internal states for $\mathcal{A}_{2P}$. Let i_a denote the a^{th} state in the concatenation i, and let $i_{a:b} = i_a \circ i_{a+1} \circ \cdots \circ i_b$, the concatenation of states i_a through i_b. Then $\frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[I_{1,t^*}=i|S_{\leq t^*}=s_{\leq t^*}\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[I_{1,t^*}=i|S_{\leq t^*}=s'_{\leq t^*}\right]}$

$$
\begin{aligned}
&= \frac{\mathbb{P}_{\mathcal{A}_{1P}}\left[I_{1,t^*-1} = i_{1:t^*-1} \mid S_{\leq t^*} = s_{\leq t^*}\right] \cdot \mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{\leq t^*} = s_{\leq t^*}, I_{2,t^*-1} = i_{t^*-1}\right]}{\mathbb{P}_{\mathcal{A}_{1P}}\left[I_{1,t^*-1} = i_{1:t^*-1} \mid S_{\leq t^*} = s'_{\leq t^*}\right] \cdot \mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{\leq t^*} = s'_{\leq t^*}, I_{2,t^*-1} = i_{t^*-1}\right]} \\
&= \frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{\leq t^*} = s_{\leq t^*}, I_{2,t^*-1} = i_{t^*-1}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{\leq t^*} = s'_{\leq t^*}, I_{2,t^*-1} = i_{t^*-1}\right]} \\
&= \frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{t^*} = s_{t^*}, I_{2,t^*-1} = i_{t^*-1}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{t^*} = s'_{t^*}, I_{2,t^*-1} = i_{t^*-1}\right]}
\end{aligned}
$$

where the second equality uses the fact that $s_{<t^*} = s'_{<t^*}$, and the third equality uses I_{2,t^*}'s conditional independence from $S_{\leq t^*-1}$ given I_{2,t^*-1}. Now, since I_{2,t^*-1} and S_{t^*} are independent, we multiply by $1 = \frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*-1}=i_{t^*-1}|S_{t^*}=s_{t^*}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*-1}=i_{t^*-1}|S_{t^*}=s'_{t^*}\right]}$ to get

$$
\frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{t^*} = s_{t^*}, I_{2,t^*-1} = i_{t^*-1}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*} \mid S_{t^*} = s'_{t^*}, I_{2,t^*-1} = i_{t^*-1}\right]} = \frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*}, I_{2,t^*-1} = i_{t^*-1} \mid S_{t^*} = s_{t^*}\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{2,t^*} = i_{t^*}, I_{2,t^*-1} = i_{t^*-1} \mid S_{t^*} = s'_{t^*}\right]} \leq e^{\varepsilon}
$$

since $\mathcal{A}_{2P}$ is ε-pan-private against two intrusions. ■

Next, we show how to convert this pan-private algorithm $\mathcal{A}_{1P}$ into an equivalent locally private algorithm $\mathcal{A}_L$.

Lemma 17 *Let $\mathcal{A}_{1P}$ be an ε-pan-private algorithm as described in Lemma 16. Then there exists a sequentially interactive ε-locally private algorithm $\mathcal{A}_L$ whose transcript distribution Π_t is identical to the $\mathcal{A}_{1P}$'s state distribution I_t at each time t.*

Proof At each time t, $\mathcal{A}_{1P}$ computes a function $\mathcal{A}_{1P}(i_{t-1}, s_t)$ of its current state and the current element in the stream and concatenates it to its current state. We define $\mathcal{A}_L$ to use $\mathcal{A}_{1P}(i_{t-1}, \cdot)$ as a randomizer, add the result $\mathcal{A}_{1P}(i_{t-1}, s_t)$ to the transcript, and continue.

$\mathcal{A}_L$ is sequentially interactive because we take a single pass through the stream. Furthermore, because $\mathcal{A}_{1P}$ is ε-pan-private and maintains all previous states, the transcript Π_t of $\mathcal{A}_L$ is an ε-differentially private function of the user data. Thus $\mathcal{A}_L$ is ε-locally private. Finally, recalling that Definition 4 defined a transcript to record not only outputs but the randomizers used as well, let Π_t^{-R} denote Π_t with the randomizers omitted. Then for any input stream s, Π_t^{-R} is distributed identically to I_t. ■

We now combine Lemma 16 and Lemma 17: any $\mathcal{A}_{2P}$ that is ε-pan-private against two intrusions yields a sequentially interactive ε-locally private $\mathcal{A}_L$ such that for any input stream s and time t, $I_{2,t}$ is distributed identically to $\Pi_t^{-R,-1}$, the most recent addition to the transcript.

$\Leftarrow$ (local to pan): Let $\mathcal{A}_L \colon \Pi \to R$ be a sequentially interactive ε-locally private protocol mapping transcripts to randomizers, and let $\mathcal{A}_\mathcal{I} \colon \mathcal{I} \times \mathcal{X} \to \mathcal{I}$ be $\mathcal{A}_{2P}$'s internal algorithm with initial state $\emptyset$. We define $\mathcal{A}_\mathcal{I}(\emptyset, x_1) = (\emptyset, \mathcal{A}_L(\emptyset), \mathcal{A}_L(\emptyset)(x_1))$ and define other internal states i by $\mathcal{A}_\mathcal{I}(i, x) = i \circ (\mathcal{A}_L(i), \mathcal{A}_L(i)(x))$, the concatenation of the existing state i and the (randomizer, output) pair $(\mathcal{A}_L(i), \mathcal{A}_L(i)(x))$. Thus $I_t = \Pi_t$ at each time t. Finally, we define the output algorithm to be the identity function $\mathcal{A}_\mathcal{O}(i) = i$.

Since $\mathcal{A}_L$ is ε-locally private, its final transcript Π is an ε-differentially private function of the stream: for any transcript realization π and neighboring streams s and s', $\frac{\mathbb{P}_{\mathcal{A}_L}[\Pi=\pi|S=s]}{\mathbb{P}_{\mathcal{A}_L}[\Pi=\pi|S=s']} \leq e^\varepsilon$. Letting I^* be a random variable for the final internal state of $\mathcal{A}_{2P}$, it follows that $\frac{\mathbb{P}_{\mathcal{A}_{2P}}[I^*=\pi|S=s]}{\mathbb{P}_{\mathcal{A}_{2P}}[I^*=\pi|S=s']} \leq e^\varepsilon$. Thus the final internal state I of $\mathcal{A}_{2P}$ is also an ε-differentially private function of the stream. Moreover, because it is a transcript, I^* includes a record of all previous internal states. Thus the additional view of any two internal states (in fact, any number of internal states) is still an ε-differentially private function of the stream: fixing times $t_1, \ldots, t_c$ and corresponding internal states $\pi_1, \ldots, \pi_c$,

$$\frac{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{t_1} = \pi_1, \ldots, I_{t_c} = \pi_c, I^* = i \mid S = s\right]}{\mathbb{P}_{\mathcal{A}_{2P}}\left[I_{t_1} = \pi_1, \ldots, I_{t_c} = \pi_c, I^* = i \mid S = s'\right]} \leq e^\varepsilon.$$

Finally, since the output of $\mathcal{A}_{2P}$ is the final state I^*, $\mathcal{A}_{2P}$ is ε-pan-private against arbitrarily many (and, in particular, two) intrusions. ■

8. Constant Separation in Uniformity Testing

Recall that Definition 9 requires success probabilities of at least $2/3$, i.e.

$$\mathbb{P}\left[\text{output uniform} \mid p = U_k\right] \geq 2/3 \text{ and } \mathbb{P}\left[\text{output uniform} \mid ||p - U_k||_{TV} \geq \alpha\right] \leq 1/3.$$

As long as we achieve constant separation, i.e. have

$$\mathbb{P}\left[\text{output uniform} \mid p = U_k\right] \geq c_1 \text{ and } \mathbb{P}\left[\text{output uniform} \mid ||p - U_k||_{TV} \geq \alpha\right] \leq c_2$$

for positive $c_1 - c_2 = \Omega(1)$, we can amplify it to a $1/3$ separation by repetition. After sufficiently many repetitions, if $p = U_k$ then the proportion of "uniform" answers will concentrate at or above c_1, and if $||p - U_k||_{TV} \geq \alpha$ it will concentrate at or below c_2. By a Chernoff bound, $r = \Omega\left(\frac{1}{(c_1-c_2)^2}\right)$ repetitions suffice to distinguish between these cases. Since this is still a constant number of repetitions, our algorithms will focus on achieving any constant separation.

9. Uniformity Testing Upper Bound Proofs

Lemma 18 *For $m = \Omega\left(\frac{k^{3/4}}{\alpha\varepsilon} + \frac{\sqrt{k}}{\alpha^2}\right)$,* SIMPLEPANTEST *is an ε-pan-private uniformity tester on m samples.*

Proof Privacy: Let t be a time in the stream, let i be a possible internal state for SIMPLEPANTEST, and let o be a possible output. Let $p_{\mathcal{I},s,t}$ be the probability density function for the internal state of SIMPLEPANTEST after the first t elements of stream s, and let $p_{\mathcal{O},s,t|i}$ be the probability density function for the output given stream s such that the internal state at time t was i. Finally, fix neighboring streams s and s'. Then to prove that SIMPLEPANTEST is ε-pan-private, it suffices to show that $\frac{p_{\mathcal{I},s,t}(i) \cdot p_{\mathcal{O},s,t|i}(o)}{p_{\mathcal{I},s',t}(i) \cdot p_{\mathcal{O},s',t|i}(o)} \leq e^{\varepsilon}$.

The final output of SIMPLEPANTEST is a deterministic function of its final internal state (after the second addition of Laplace noise). The final internal state is after m samples, so it is enough to choose arbitrary internal states i_1 and i_2 and show

$$\frac{p_{\mathcal{I},s,t}(i_1) \cdot p_{\mathcal{I},s,m,t|i_1}(i_2)}{p_{\mathcal{I},s',t}(i_1) \cdot p_{\mathcal{I},s',m,t|i_1}(i_2)} \leq e^{\varepsilon}. \tag{3}$$

We first recall a basic fact about differential privacy: if f is a real-valued function with sensitivity Δf, i.e. a function whose output changes by at most Δ between neighboring databases, then adding $\mathsf{Lap}\left(\frac{\Delta f}{\varepsilon}\right)$ noise to the output of f is ε-differentially private (see e.g. Theorem 3.4 in the survey of Dwork et al. (2014)). Here, each bin of H is a 1-sensitive function and each sample alters a single bin. Thus by the first application of $\mathsf{Lap}\left(\frac{1}{\varepsilon}\right)$ noise to each bin we get $\frac{p_{\mathcal{I},s,t}(i_1)}{p_{\mathcal{I},s',t}(i_1)} \leq e^{\varepsilon}$. Similarly, the second application of $\mathsf{Lap}\left(\frac{1}{\varepsilon}\right)$ noise to each bin implies $\frac{p_{\mathcal{I},s,m,t|i_1}(i_2)}{p_{\mathcal{I},s',m,t|i_1}(i_2)} \leq e^{\varepsilon}$. To get the overall claim, we split into two cases. If $s_{\leq t} = s'_{\leq t}$, then $\frac{p_{\mathcal{I},s,t}(i_1)}{p_{\mathcal{I},s',t}(i_1)} = 1$. If instead $s_{\leq t} \neq s'_{\leq t}$, then $s_{>t} = s'_{>t}$, so $\frac{p_{\mathcal{I},s,m,t|i_1}(i_2)}{p_{\mathcal{I},s',m,t|i_1}(i_2)} = 1$. Thus Equation 3 holds.

Sample complexity: To better analyze Z', we decompose it as the sum of a non-private χ^2-statistic Z and a noise term Y,

$$Z = \sum_{i=1}^{k} \frac{(N_i - m/k)^2 - N_i}{m/k} \text{ and } Y = \sum_{i=1}^{k} \frac{[Y_i + Y'_i]^2 + 2[Y_i + Y'_i](N_i - m/k) - [Y_i + Y'_i]}{m/k}.$$

where N_i is the true stream count of item i and $Y_i, Y'_i \sim \mathsf{Lap}\left(\frac{1}{\varepsilon}\right)$ are the first and second addition of Laplace noise. This lets us rewrite $Z' = Z + Y$. In the uniform case, we will give a high-probability upper bound for $Z + Y$, and in the non-uniform case we will give a high-probability lower bound. Fortunately, Acharya et al. (2015) prove several results about Z. We summarize these results in Lemma 19.

Lemma 19 (Lemmas 2 and 3 from Acharya et al. (2015)) *If $p = U_k$ and $m = \Omega\left(\frac{\sqrt{k}}{\alpha^2}\right)$, then $\mathbb{E}[Z] \leq \frac{\alpha^2 m}{500}$ and $Var[Z] \leq \frac{\alpha^4 m^2}{500000}$. If $||p - U_k||_{TV} \geq \alpha$, then $\mathbb{E}[Z] \geq \frac{\alpha^2 m}{5}$ and $Var[Z] \leq \frac{\mathbb{E}[Z]^2}{100}$.*

We split into cases depending on p. For each case, Lemma 19 will control Z, and our task will be to control Y.

Case 1: $p = U_k$. By Lemma 19, $\mathbb{E}[Z] \leq \frac{\alpha^2 m}{500}$ and $\mathrm{Var}[Z] \leq \frac{\alpha^4 m^2}{500000}$. By Chebyshev's inequality, $\mathbb{P}\left[Z > \left(\frac{1}{500} + \frac{c}{500\sqrt{2}}\right)\alpha^2 m\right] \leq \frac{1}{c^2}$.

Turning our attention to Y, define

$$A = \sum_{i=1}^{k} \frac{[Y_i + Y_i']^2}{m/k},\ B = \sum_{i=1}^{k} \frac{2[Y_i + Y_i'](N_i - m/k)}{m/k}, \text{ and } C = \sum_{i=1}^{k} \frac{Y_i + Y_i'}{m/k}.$$

Then we can rewrite $Y = A + B - C$. We control each of A, B, and C in turn. First, by the independence of all draws of noise, $\mathbb{E}[A] = \frac{k^2 \mathbb{E}\left[[Y_i+Y_i']^2\right]}{m} = \frac{2k^2 \mathrm{Var}[Y_i]}{m} = \frac{4k^2}{\varepsilon^2 m}$ because $\mathrm{Var}\left[\mathsf{Lap}\left(\frac{1}{\varepsilon}\right)\right] = \frac{2}{\varepsilon^2}$. Next,

$$\begin{aligned}
\mathrm{Var}[A] &= \frac{k^3}{m^2}\mathrm{Var}\left[Y_i^2 + 2Y_iY_i' + Y_i'^2\right] \\
&= \frac{k^3}{m^2}\left(\mathbb{E}\left[(Y_i^2 + 2Y_iY_i' + Y_i'^2)^2\right] - \mathbb{E}\left[Y_i^2 + 2Y_iY_i' + Y_i'^2\right]^2\right) \\
&= \frac{k^3}{m^2}\left(\left[2\mathbb{E}\left[Y_i^4\right] + 6\mathbb{E}\left[Y_i^2\right]^2\right] - 4\mathbb{E}\left[Y_i^2\right]^2\right) \\
&= \frac{2k^3}{m^2}\left(\mathbb{E}\left[Y_i^4\right] + \mathbb{E}\left[Y_i^2\right]^2\right) \\
&= \frac{2k^3}{m^2}\left(\frac{12}{\varepsilon^4} + \frac{4}{\varepsilon^4}\right) \\
&= \frac{32k^3}{\varepsilon^4 m^2}
\end{aligned}$$

where we use $\mathbb{E}\left[Y_i^4\right] = \frac{\varepsilon}{2}\int_0^\infty x^4 e^{-\varepsilon x}dx = \frac{12}{\varepsilon^4}$ by repeated integration by parts. With Chebyshev's inequality, $\mathbb{P}\left[A > \frac{4k^2}{\varepsilon^2 m} + 6c\frac{k^{3/2}}{\varepsilon^2 m}\right] < \frac{1}{c^2}$.

To bound B, we use $\mathbb{E}[B] = 0$ and

$$\begin{aligned}
\mathrm{Var}[B] &= \frac{4k^2}{m^2} \cdot \mathrm{Var}\left[\sum_{i=1}^{k}[Y_i + Y_i']\left(N_i - \frac{m}{k}\right)\right] \\
&= \frac{4k^2}{m^2} \cdot \mathbb{E}\left[\left(\sum_{i=1}^{k}[Y_i + Y_i']\left[N_i - \frac{m}{k}\right]\right)^2\right] \\
&= \frac{4k^2}{m^2} \sum_{i_1, i_2 \in [k]} \mathbb{E}\left[(Y_{i_1} + Y_{i_1}')(Y_{i_2} + Y_{i_2}')\right] \cdot \mathbb{E}\left[\left(N_{i_1} - \frac{m}{k}\right)\left(N_{i_2} - \frac{m}{k}\right)\right] \\
&= \frac{4k^2}{m^2} \sum_{i=1}^{k} \mathbb{E}\left[(Y_i + Y_i')^2\right] \cdot \mathbb{E}\left[\left(N_i - \frac{m}{k}\right)^2\right] \\
&= \frac{16k^3}{\varepsilon^2 m^2}\left(\mathbb{E}\left[N_1^2\right] - \frac{2m\mathbb{E}[N_1]}{k} + \frac{m^2}{k^2}\right) \\
&= \frac{16k^3}{\varepsilon^2 m^2}\left(\mathrm{Var}[N_1] + \mathbb{E}[N_1]^2 - \frac{2m^2}{k^2} + \frac{m^2}{k^2}\right) \\
&= \frac{16k^2}{\varepsilon^2 m}
\end{aligned}$$

where the last two equalities use $N_i \sim \mathsf{Poisson}\left(\frac{m}{k}\right)$ and $\mathsf{Var}\left[\mathsf{Poisson}\left(\frac{m}{k}\right)\right] = \frac{m}{k}$. Again applying Chebyshev's inequality gives $\mathbb{P}\left[B > 4c\frac{k}{\varepsilon\sqrt{m}}\right] < \frac{1}{c^2}$.

Similarly, $\mathbb{E}\left[C\right] = 0$, and with $\mathsf{Var}\left[C\right] = \frac{k^3}{m^2} \cdot \mathsf{Var}\left[Y_i + Y_i'\right] = \frac{4k^3}{\varepsilon^2 m^2}$, $\mathbb{P}\left[C < -2c\frac{k^{3/2}}{\varepsilon m}\right] \leq \frac{1}{c^2}$.

Combining the above bounds on Z, A, B, and C, with probability at least $1 - \frac{4}{c^2}$,

$$Z' \leq \left(\tfrac{1}{500} + \tfrac{c}{500\sqrt{2}}\right)\alpha^2 m + \frac{4k^2}{\varepsilon^2 m} + 6c\frac{k^{3/2}}{\varepsilon^2 m} + 4c\frac{k}{\varepsilon\sqrt{m}} + 2c\frac{k^{3/2}}{\varepsilon m}.$$

Taking $c = 4\sqrt{2}$ and

$$T_U = \tfrac{1}{100}\alpha^2 m + 4\tfrac{k^2}{\varepsilon^2 m} + 24\sqrt{2}\tfrac{k^{3/2}}{\varepsilon^2 m} + 16\sqrt{2}\tfrac{k}{\varepsilon\sqrt{m}} + 8\sqrt{2}\tfrac{k^{3/2}}{\varepsilon m},$$

$\mathbb{P}\left[Z' \leq T_U\right] \geq 7/8$.

Case 2: $||p - U_k||_{TV} \geq \alpha$. By Lemma 19, $\mathbb{E}\left[Z\right] \geq \frac{\alpha^2 m}{5}$ and $\mathsf{Var}\left[Z\right] \leq \frac{\mathbb{E}[Z]^2}{100}$. Chebyshev's inequality now gives

$$1 - \frac{1}{c^2} \leq \mathbb{P}\left[Z \geq \mathbb{E}\left[Z\right] - c\sqrt{\mathsf{Var}\left[Z\right]}\right] \leq \mathbb{P}\left[Z \geq \left(1 - \frac{c}{10}\right)\mathbb{E}\left[Z\right]\right] \leq \mathbb{P}\left[Z \geq \left(1 - \frac{c}{10}\right)\frac{\alpha^2 m}{5}\right]$$

where the last inequality requires $c \leq 10$. Returning to the decomposition of Y used in Case 1, A and C are unchanged and we can use our previous expressions for them (with appropriate sign changes for lower bounds). Our last task is to lower bound $B = \frac{2k}{m}\sum_{i=1}^k [Y_i + Y_i'](N_i - m/k)$. For any term i, Y_i and Y_i' are symmetric, so

$$\mathbb{P}\left[[Y_i + Y_i'](N_i - m/k) > 0\right] = \mathbb{P}\left[[Y_i + Y_i'](N_i - m/k) < 0\right]$$

and $\mathbb{P}\left[B \geq 0\right] \geq 1/2$.

Summing up, with probability at least $\frac{1}{2} - \frac{3}{c'^2}$,

$$Z' \geq \left(\frac{1}{5} - \frac{c'}{50}\right)\alpha^2 m + 4\frac{k^2}{\varepsilon^2 m} - 6c'\frac{k^{3/2}}{\varepsilon^2 m} - 2c'\frac{k^{3/2}}{\varepsilon m}.$$

Taking $c' = 2\sqrt{3}$ and $T_\alpha = \frac{\alpha^2 m}{10} + 4\frac{k^2}{\varepsilon^2 m} - 12\sqrt{3}\frac{k^{3/2}}{\varepsilon^2 m} - 4\sqrt{3}\frac{k^{3/2}}{\varepsilon m}$, $\mathbb{P}\left[Z' \geq T_\alpha\right] \geq \frac{1}{4}$.

For $T_\alpha > T_U$, it is enough that $T_\alpha - T_U > 0$.

$$T_\alpha - T_U = \frac{9}{100}\alpha^2 m - \left(12\sqrt{3} + 24\sqrt{2}\right)\frac{k^{3/2}}{\varepsilon^2 m} - 16\sqrt{2}\frac{k}{\varepsilon\sqrt{m}} - \left(4\sqrt{3} + 8\sqrt{2}\right)\frac{k^{3/2}}{\varepsilon m}.$$

Dropping constants, we need $\alpha^2 m = \Omega\left(\frac{k^{3/2}}{\varepsilon^2 m} + \frac{k}{\varepsilon\sqrt{m}} + \frac{k^{3/2}}{\varepsilon m}\right)$. We can drop the lower-order term $\frac{k^{3/2}}{\varepsilon m}$ and get $\alpha^2 m = \Omega\left(\frac{k^{3/2}}{\varepsilon^2 m} + \frac{k}{\varepsilon\sqrt{m}}\right)$, i.e. $m = \Omega\left(\frac{k^{3/4}}{\alpha\varepsilon} + \frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}}\right)$.

Putting it all together and recalling the assumption from Lemma 19, there exists constant c such that if $m > c\left(\frac{k^{3/4}}{\alpha\varepsilon} + \frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2}\right)$ then

$$\mathbb{P}\left[\text{output “uniform”} \mid ||p - U_k||_{TV} \geq \alpha\right] \leq 3/4 \text{ and } \mathbb{P}\left[\text{output “uniform”} \mid p = U_k\right] \geq 7/8.$$

Thus we get a constant $1/8$ separation. By the amplification argument outlined after Definition 9, SIMPLEPANTEST is a uniformity tester. Finally,

$$\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} = \left(\frac{k^{3/4}}{\alpha\varepsilon}\right)^{2/3} \cdot \left(\frac{\sqrt{k}}{\alpha^2}\right)^{1/3} \leq \frac{2}{3}\left(\frac{k^{3/4}}{\alpha\varepsilon}\right) + \frac{1}{3}\left(\frac{\sqrt{k}}{\alpha^2}\right)$$

by the AM-GM inequality, and our statement simplifies to $m = \Omega\left(\frac{k^{3/4}}{\alpha\varepsilon} + \frac{\sqrt{k}}{\alpha^2}\right)$. ■

Lemma 20 *Let p be a distribution over $[k]$ such that $||p - U_k||_{TV} = \alpha$ and let $G_1, \ldots, G_n$ be a uniformly random partition of $[k]$ into $n > 1$ subsets of size $\Theta(k/n)$. Define induced distribution p_n over $[n]$ by $p_n(j) = \sum_{i \in G_j} p(i)$ for each $j \in [n]$. Then, with probability $\geq \frac{1}{954}$ over the selection of $G_1, \ldots, G_n$,*

$$||p_n - U_n||_{TV} = \Omega\left(\alpha\sqrt{\frac{n}{k}}\right).$$

Proof It is equivalent to sample $G_1, \ldots, G_n$ as follows: randomly partition $[k]$ into $n/2$ same-size subsets $G'_1, \ldots, G'_{n/2}$ (for neatness, we assume n is even), and then randomly halve each of those to produce G_1 and G_2 (from G'_1), G_3 and G_4 (from G'_2), and so on. We use the following lemma from Acharya et al. (2019a) to connect the distances induced by $\{G'_a\}_{a=1}^{n/2}$ and $\{G_b\}_{b=1}^{n}$. Here, for a set S we let $p(S)$ denote the total probability mass of set S, $p(S) = \sum_{s \in S} p(s)$.

Lemma 21 (Corollary 15 in Acharya et al. (2019a)) *Let p be a distribution over $[k]$ with $||p - U_k||_{TV} \geq \alpha$, and let U be a random subset of $[k]$ of size $k/2$. Then $\mathbb{P}_U\left[|p(U) - 1/2| \geq \frac{\alpha}{\sqrt{5k}}\right] > \frac{1}{477}$.*

Slightly more generally, the proof of Lemma 21 shows that for any distribution p over $[k]$ and $S \subset [k]$, if $\frac{1}{2}\sum_{i \in S}|p(i) - \frac{1}{k}| \geq \alpha'$, and we choose a random subset $S' \subset S$ of size $\frac{|S|}{2}$, then $\mathbb{P}_{S'}\left[|p(S') - \frac{p(S)}{2}| \geq \frac{\alpha'}{\sqrt{5|S|}}\right] > \frac{1}{477}$.

Fix the choice of $G'_1, \ldots, G'_{n/2}$. For each $a \in [n/2]$, let $\alpha_a = \frac{1}{2}\sum_{i \in G'_a}|p(i) - \frac{1}{k}|$, the portion of $||p - U_k||_{TV}$ contributed by G'_a. Replacing α' with α_a and $|S|$ with $k/(n/2)$ above, for each $a \in [n/2]$,

$$\mathbb{P}\left[\left|p(G_{2a-1}) - \frac{p(G'_a)}{2}\right| \geq \alpha_a\sqrt{\frac{n}{10k}}\right] \geq \frac{1}{477}.$$

$p(G_{2a-1}) + p(G_{2a}) = p(G'_a)$, so

$$\mathbb{P}\left[|p(G_{2a-1}) - p(G_{2a})| \geq 2\alpha_a\sqrt{\frac{n}{10k}}\right] \geq \frac{1}{477}.$$

Then by triangle inequality

$$\mathbb{P}\left[\left|p(G_{2a-1}) - \frac{1}{n}\right| + \left|p(G_{2a}) - \frac{1}{n}\right| \geq 2\alpha_a\sqrt{\frac{n}{10k}}\right] \geq \frac{1}{477}$$

and in particular

$$\mathbb{E}\left[\left|p(G_{2a-1}) - \frac{1}{n}\right| + \left|p(G_{2a}) - \frac{1}{n}\right|\right] \geq \frac{2\alpha_a}{477}\sqrt{\frac{n}{10k}}.$$

For each $b \in [n]$ define $Y_b = \min\left(\left|p(G_b) - \frac{1}{n}\right|, \alpha_{\lceil b/2 \rceil}\sqrt{\frac{n}{10k}}\right)$. Let $Y = \sum_{b=1}^n Y_b$. First, we can lower bound $\mathbb{E}[Y]$, over the choice of $G'_1, \dots, G'_{n/2}$ and $G_1, \dots, G_n$, as

$$\begin{aligned}\mathbb{E}[Y] &= \sum_{b=1}^n \mathbb{E}\left[\min\left(\left|p(G_b) - \frac{1}{n}\right|, \alpha_{\lceil b/2 \rceil}\sqrt{\frac{n}{10k}}\right)\right] \\ &\geq \sum_{b=1}^n \tfrac{\alpha_{\lceil b/2 \rceil}}{477}\sqrt{\tfrac{n}{10k}} \\ &= \tfrac{2\alpha}{477}\sqrt{\tfrac{n}{10k}} \qquad (4)\end{aligned}$$

where the inequality uses the expectation lower bound above.

Second, by definition of Y_b, $\max(Y) \leq \sum_{b=1}^n \alpha_{\lceil b/2 \rceil}\sqrt{\frac{n}{10k}} = 2\alpha\sqrt{\frac{n}{10k}}$. Now assume for contradiction that $\mathbb{P}\left[Y \geq \frac{\alpha}{477}\sqrt{\frac{n}{10k}}\right] < \frac{1}{954}$. Then

$$\mathbb{E}[Y] < \tfrac{\alpha}{477}\sqrt{\tfrac{n}{10k}} + \tfrac{\max(Y)}{954} \leq \tfrac{2\alpha}{477}\sqrt{\tfrac{n}{10k}}.$$

Thus $\mathbb{E}[Y] < \frac{2\alpha}{477}\sqrt{\frac{n}{10k}}$, which contradicts Equation 4. It follows that our assumption is false, and $\mathbb{P}\left[Y \geq \frac{\alpha}{477}\sqrt{\frac{n}{10k}}\right] \geq \frac{1}{954}$. The final claim follows from

$$\begin{aligned}\frac{Y}{2} &= \frac{1}{2}\sum_{b=1}^n \min\left(\left|p(G_b) - \frac{1}{n}\right|, \alpha_{\lceil b/2 \rceil}\sqrt{\frac{n}{10k}}\right) \\ &\leq \frac{1}{2}\sum_{b=1}^n \left|p(G_b) - \frac{1}{n}\right| \\ &= ||p_n - U_n||_{TV}.\end{aligned}$$

■

Theorem 22 *For $m = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2} + \frac{\sqrt{k}}{\alpha\varepsilon}\right)$,* PANTEST *is an ε-pan-private uniformity tester on m samples.*

Proof Privacy: PANTEST only interacts with the data through SIMPLEPANTEST, so PANTEST inherits SIMPLEPANTEST's pan-privacy guarantee.

Sample complexity: Substituting n for k and $\alpha\sqrt{\frac{n}{k}}$ for α in Lemma 10, we require

$$m = \Omega\left(\frac{n^{1/4}\sqrt{k}}{\alpha\varepsilon} + \frac{k}{\alpha^2\sqrt{n}}\right). \qquad (5)$$

We consider the three cases for $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}}$. Together, these cases exhaust the possible relationships among α, k, and ε, with a different highest-order term in each. This leads to the three terms in our bound.

First, if $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} \in [2, k]$, then $n = \lfloor \frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} \rfloor$. By Equation 5 it is enough for

$$m = \Omega\left(\frac{k^{1/6}\varepsilon^{1/3}\sqrt{k}}{\alpha^{1/3}\alpha\varepsilon} + \frac{k}{\alpha^2 \cdot \frac{k^{1/3}\varepsilon^{2/3}}{\alpha^{2/3}}}\right) = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}}\right).$$

Next, if $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} > k$, then $n = k$, and Equation 5 necessitates $m = \Omega\left(\frac{k^{3/4}}{\alpha\varepsilon} + \frac{\sqrt{k}}{\alpha^2}\right)$. The condition $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} > k$ gives $\frac{\varepsilon^4}{\alpha^4} > k$, so $\frac{\varepsilon}{\alpha} > k^{1/4}$, and then multiplying both sides by $\frac{\sqrt{k}}{\alpha\varepsilon}$ gives $\frac{\sqrt{k}}{\alpha^2} > \frac{k^{3/4}}{\alpha\varepsilon}$.Thus it suffices for $m = \Omega\left(\frac{\sqrt{k}}{\alpha^2}\right)$.

Finally, if $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} < 2$, then $n = 2$ and by Equation 5 we require $m = \Omega\left(\frac{\sqrt{k}}{\alpha\varepsilon} + \frac{k}{\alpha^2}\right)$. $\frac{k^{2/3}\varepsilon^{4/3}}{\alpha^{4/3}} < 2$ implies $\varepsilon < \frac{2\alpha}{\sqrt{k}}$, so multiplying both sides by $\frac{k}{\alpha^2\varepsilon}$ yields $\frac{k}{\alpha^2} < \frac{2\sqrt{k}}{\alpha\varepsilon}$ and $\frac{\sqrt{k}}{\alpha\varepsilon} = \Omega\left(\frac{k}{\alpha^2}\right)$. Thus it suffices for $m = \Omega\left(\frac{\sqrt{k}}{\alpha\varepsilon}\right)$. ■

10. Uniformity Testing Lower Bound Proofs

Theorem 23 *For $\varepsilon = O(1)$ and $\alpha \leq 1/2$, any ε-pan-private uniformity tester requires $m = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{\sqrt{k}}{\alpha^2} + \frac{1}{\alpha\varepsilon}\right)$ samples.*

Proof First, recall the centrally private lower bound (Acharya et al. (2018)):

$$m = \Omega\left(\frac{\sqrt{k}}{\alpha^2} + \frac{\sqrt{k}}{\alpha\sqrt{\varepsilon}} + \frac{k^{1/3}}{\alpha^{4/3}\varepsilon^{2/3}} + \frac{1}{\alpha\varepsilon}\right).$$

We will prove $m = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}}\right)$ in the pan-private case. $\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}}$ dominates the third term above and also dominates the second term for $\varepsilon = O(1)$, so this produces our final lower bound.

We start with the lower bound construction used by Diakonikolas et al. (2019), which itself uses the Paninski lower bound construction (Paninski (2008)). Let X be a uniform random bit determining which of two distributions over $[2k]$ generates the samples. For both $X = 0$ and $X = 1$ we draw $Y_1, \ldots, Y_k \in \{\pm 1\}$ i.i.d. uniformly at random. If $X = 0$, $p = U_{2k}$. If instead $X = 1$, then we pair the bins as $\{1, 2\}, \{3, 4\}, \ldots, \{2k - 1, 2k\}$ and define $p(2j - 1) = \frac{1+Y_j\alpha}{2k}$ and $p(2j) = \frac{1-Y_j\alpha}{2k}$. Thus if $X = 0$ then p is uniform, and if $X = 1$ each pair i of bins is biased toward one of the bins according to Y_j. Equivalently, we can view each sample $S_t \sim p$ as a pair (J_t, V_t) where $J_t \in [k]$ determines the bin pair chosen and $V_t \in \{0, 1\}$ determines which of the bin pair is chosen. Thus $J_t \sim U_k$, and $V_t \sim \text{Ber}\left(\frac{1}{2}\right)$ if $X = 0$ or $V_t \sim \text{Ber}\left([1 + \alpha Y_{j_t}]/2\right)$ if $X = 1$, where $\text{Ber}(\cdot)$ denotes the Bernoulli distribution.

To avoid confusion with the mutual information $I(\cdot)$, denote by M_t the random variable for the internal state of the algorithm after seeing sample S_t. Our goal is to upper bound the mutual information between X and the internal state after m samples,

$$\begin{aligned} I(X; M_m) &= \sum_{t=1}^{m} I(X; M_t) - I(X; M_{t-1}) \\ &\leq \sum_{t=1}^{m} I(X; M_{t-1}, S_t) - I(X; M_{t-1}) \\ &= \sum_{t=1}^{m} I(X; S_t \mid M_{t-1}) \\ &= \sum_{t=1}^{m} I(X; V_t \mid M_{t-1}, J_t) \end{aligned} \tag{6}$$

where the last equality uses $S_t = (J_t, V_t)$ and the independence of X and J_t

We now have a narrower goal: we choose an arbitrary term in the sum in Equation (6) and upper bound it. For neatness, we use the convention that $H_2(p)$ is the entropy of a $\mathsf{Ber}\left(p\right)$ random variable. When subscripting we abuse notation and let $a \sim A$ denote a sample a from the distribution for random variable A. The following reproduces (and slightly expands) the first part of the argument given by Diakonikolas et al. (2019). It largely reduces to rewriting mutual information in terms of binary entropy and expanding conditional probabilities.

We start by rewriting the chosen term $I(X; V_t \mid M_{t-1}, J_t)$ as

$$\begin{aligned} &= \mathbb{E}_{m^* \sim M_{t-1}}\left[\mathbb{E}_{j \sim J_t}\left[H(V_t \mid M_{t-1} = m^*, J_t = j)\right]\right] \\ &\quad - \mathbb{E}_{m^* \sim M_{t-1}}\left[\mathbb{E}_{j \sim J_t}\left[\mathbb{E}_{x \sim X}\left[H(V_t \mid M_{t-1} = m^*, J_t = j, X = x)\right]\right]\right] \\ &= \mathbb{E}_{m^* \sim M_{t-1}}\left[\mathbb{E}_{j \sim J_t}\left[H_2(\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j\right])\right]\right] \\ &\quad - \mathbb{E}_{m^* \sim M_{t-1}}\left[\mathbb{E}_{j \sim J_t}\left[\mathbb{P}\left[X = 1 \mid M_{t-1} = m^*, J_t = j\right] H_2(\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j, X = 1\right])\right]\right] \\ &\quad - \mathbb{E}_{m^* \sim M_{t-1}}\left[\mathbb{E}_{j \sim J_t}\left[\mathbb{P}\left[X = 0 \mid M_{t-1} = m^*, J_t = j\right] H_2(\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j, X = 0\right])\right]\right] \end{aligned}$$

where the second equality uses $H_2(p) = H_2(1-p)$. Let $\beta_{t-1}^{m^*,j} = \mathbb{P}\left[X = 1 \mid M_{t-1} = m^*, J_t = j\right]$. Since J_t is a uniform draw from $[k]$ independent of M_{t-1}, we now continue the above chain of equalities as

$$\begin{aligned} &= \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k} H_2\left(\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j\right]\right)\right] \\ &\quad - \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k} \beta_{t-1}^{m^*,j} H_2\left(\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j, X = 1\right]\right)\right] \\ &\quad - \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k} (1 - \beta_{t-1}^{m^*,j}) H_2\left(\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j, X = 0\right]\right)\right]. \end{aligned} \tag{7}$$

Now recall that $V_t \sim \mathsf{Ber}\left(\frac{1}{2}\right)$ when $X = 0$ and $V_t \sim \mathsf{Ber}\left([1 + \alpha Y_{J_t}]/2\right)$ when $X = 1$. Then we can rewrite $\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j\right]$ as

$$\begin{aligned}
&= \beta_{t-1}^{m^*,j}\mathbb{P}\left[V_t = 0 \mid X = 1, M_{t-1} = m^*, J_t = j\right] + (1 - \beta_{t-1}^{m^*,j})\mathbb{P}\left[V_t = 0 \mid X = 0, M_{t-1} = m^*, J_t = j\right] \\
&= \beta_{t-1}^{m^*,j}\mathbb{P}\left[V_t = 0 \mid X = 1, M_{t-1} = m^*, J_t = j, Y_j = 1\right]\mathbb{P}\left[Y_j = 1 \mid M_{t-1} = m^*\right] \\
&\quad + \ \beta_{t-1}^{m^*,j}\mathbb{P}\left[V_t = 0 \mid X = 1, M_{t-1} = m^*, J_t = j, Y_j = -1\right]\mathbb{P}\left[Y_j = -1 \mid M_{t-1} = m^*\right] \\
&\quad + \ (1 - \beta_{t-1}^{m^*,j})\mathbb{P}\left[V_t = 0 \mid X = 0\right] \\
&= \beta_{t-1}^{m^*,j}\left(\mathbb{P}\left[Y_j = 1 \mid M_{t-1} = m^*\right] \cdot \frac{1 - \alpha}{2} + \mathbb{P}\left[Y_j = -1 \mid M_{t-1} = m^*\right] \cdot \frac{1 + \alpha}{2}\right) + \frac{1 - \beta_{t-1}^{m^*,j}}{2} \\
&= \beta_{t-1}^{m^*,j}\mathbb{E}\left[\frac{1 - \alpha Y_j}{2} \mid M_{t-1} = m^*\right] + \frac{1 - \beta_{t-1}^{m^*,j}}{2} \\
&= \frac{\beta_{t-1}^{m^*,j}(1 - \alpha\mathbb{E}\left[Y_j \mid M_{t-1} = m^*\right]}{2} + \frac{1 - \beta_{t-1}^{m^*,j}}{2} = \frac{1 - \alpha\beta_{t-1}^{m^*,j}\mathbb{E}\left[Y_j \mid M_{t-1} = m^*\right]}{2}.
\end{aligned}$$

where the first equality uses the independence of Y_j from X and J_t as well as the independence of V_t from M_{t-1} and J_t conditioned on $X = 0$, and the second equality uses the independence of V_t and M_{t-1} conditioned on X, $J_t = j$, and Y_j. Thus

$$\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j\right] = \frac{1 - \alpha\beta_{t-1}^{m^*,j}\mathbb{E}\left[Y_j \mid M_{t-1} = m^*\right]}{2}.$$

Using the work above, we can also rewrite

$$\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j, X = 1\right] = \frac{1 - \alpha\mathbb{E}\left[Y_j \mid M_{t-1} = m^*\right]}{2}$$

and

$$\mathbb{P}\left[V_t = 0 \mid M_{t-1} = m^*, J_t = j, X = 0\right] = \frac{1}{2}.$$

In the following chain of equalities, for space we let E be the event that $M_{t-1} = m^*$. Now we can return to Equation 7 and, since $H_2(\frac{1}{2}) = 1$, get

$$\begin{aligned}
(7) &= \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k}\left(H_2\left(\frac{1 - \alpha\beta_{t-1}^{m^*,j}\mathbb{E}\left[Y_j \mid E\right]}{2}\right) - \beta_{t-1}^{m^*,j}H_2\left(\frac{1 - \alpha\mathbb{E}\left[Y_j \mid E\right]}{2}\right) - (1 - \beta_{t-1}^{m^*,j})\right)\right] \\
&= \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k}\left(\beta_{t-1}^{m^*,j}\left[1 - H_2\left(\frac{1 - \alpha\mathbb{E}\left[Y_j \mid E\right]}{2}\right)\right] - \left[1 - H_2\left(\frac{1 - \alpha\beta_{t-1}^{m^*,j}\mathbb{E}\left[Y_j \mid E\right]}{2}\right)\right]\right)\right] \\
&\leq \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k}\left[1 - H_2\left(\frac{1 - \alpha\mathbb{E}\left[Y_j \mid E\right]}{2}\right)\right]\right] \\
&= \mathbb{E}_{m^* \sim M_{t-1}}\left[\frac{1}{k}\sum_{j=1}^{k}\left[1 - H_2\left(\frac{1 + \alpha\mathbb{E}\left[Y_j \mid E\right]}{2}\right)\right]\right] \qquad (8)
\end{aligned}$$

where the inequality uses $H_2, \beta_{t-1}^{m^*,j} \leq 1$ and the equality uses $H_2\left(\frac{1}{2} - b\right) = H_2\left(\frac{1}{2} + b\right)$. We now control the terms with H_2. The Taylor series for $H_2(p)$ near $1/2$ is $H_2(p) = 1 - \frac{1}{2\ln(2)} \sum_{n=1}^{\infty} \frac{(1-2p)^{2n}}{n(2n-1)}$, so for $a < 1/2$

$$1 - H_2\left(\frac{1}{2} + a\right) < \sum_{n=1}^{\infty} \frac{(2a)^{2n}}{n^2} = 4a^2 \sum_{n=1}^{\infty} \frac{(2a)^{2n-2}}{n^2} < 4a^2 \sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{2a^2\pi^2}{3}.$$

Substituting $1 - H_2\left(\frac{1}{2} + a\right) < \frac{2\pi^2 a^2}{3}$ into Inequality 8 and tracing back to Equation 6,

$$I(X; V_t \mid M_{t-1}, J_t) < \frac{\pi^2\alpha^2}{6k} \mathbb{E}_{m^* \sim M_{t-1}} \left[\sum_{j=1}^{k} \mathbb{E}\left[Y_j \mid M_{t-1} = m^*\right]^2\right] \tag{9}$$

We now depart from the argument of Diakonikolas et al. (2019). Our new goal is to upper bound

$$\begin{aligned} A &= \mathbb{E}_{m^* \sim M_{t-1}} \left[\sum_{j=1}^{k} \mathbb{E}\left[Y_j \mid M_{t-1} = m^*\right]^2\right] \\ &= \mathbb{E}_{m^* \sim M_{t-1}} \left[\sum_{j=1}^{k} \left(2\mathbb{P}\left[Y_j = 1 \mid M_{t-1} = m^*\right] - 1\right)^2\right] \\ &= \mathbb{E}_{m^* \sim M_{t-1}} \left[\sum_{j=1}^{k} \left(\frac{\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right]}{\mathbb{P}\left[M_{t-1} = m^*\right]} - 1\right)^2\right] \end{aligned}$$

by Bayes' rule and $\mathbb{P}\left[Y_j = 1\right] = 1/2$. To upper bound this sum, we choose an arbitrary j and show that $\frac{\mathbb{P}[M_{t-1}=m^* \mid Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]}$ is close to 1. We pause to recap what we've accomplished and what remains. Note that proving $\frac{\mathbb{P}[M_{t-1}=m^* \mid Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]} \approx 1$ "looks like" a privacy statement: we are claiming that the state distribution M_{t-1} looks similar when its input distribution is slightly different. However, there is still a gap between a difference in input distribution and a difference in input. We close this gap in the following lemma, which relies on pan-privacy.

Lemma 24 $\left|\frac{\mathbb{P}[M_{t-1}=m^* \mid Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]} - 1\right| = O\left(\frac{\alpha\varepsilon t}{k}\right)$.

Proof We will prove this claim by showing that both the numerator and denominator of $\frac{\mathbb{P}[M_{t-1}=m^* \mid Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]}$ fall into a bounded range. This implies that the whole fraction is near 1.

First consider the case $X = 0$. Then the Y_j are irrelevant, so $\left|\frac{\mathbb{P}[M_{t-1}=m^* \mid Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]} - 1\right| = 0$.

Next, consider the case $X = 1$. It will be useful to consider an equivalent method of sampling the stream S. At each time step t, we first sample a bin pair $J_t \sim_U [k]$ uniformly at random from the k bin pairs. Having sampled bin pair j, with probability $1 - \alpha$ we take a uniform random draw from $\{2j - 1, 2j\}$. With the remaining probability α, if $Y_j = 1$ then we sample $2j - 1$, and if $Y_j = -1$ then we sample $2j$. Note that this method is equivalent because if $Y_j = 1$ then $\mathbb{P}\left[\text{sample } 2j - 1\right] = \frac{1}{k} \cdot \frac{1-\alpha}{2} + \frac{\alpha}{k} = \frac{1+\alpha}{2k}$ and $\mathbb{P}\left[\text{sample } 2j\right] = \frac{1-\alpha}{2k}$, with these equalities swapped for $Y_j = -1$. With this view of sampling, let $E_{j,t}^{\alpha} = 1$ if $J_t = j$ and we sample from the α mixture component and $E^\alpha_{j,t} = 0$ otherwise. Finally, let $N^\alpha_{j,t} = \sum_{t'=1}^{t} E^\alpha_{j,t'}$, the number of samples from the α mixture component of bin pair j through the first t stream elements.

We pause to justify bothering with this alternate view. We use it because the original ratio $\frac{\mathbb{P}[M_{t-1}=m^* | Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]}$ is comparing the views of M_{t-1} depending on Y_j. It is not obvious how to directly use pan-privacy to reason about this comparison because Y_j is a property of the distribution generating the samples (stream elements) rather than the samples themselves. In contrast, pan-privacy is a guarantee formulated in terms of the samples. By defining the $E^\alpha_{j,t}$ and $N^\alpha_{j,t}$ above we better connect Y_j to the actual samples received. The alternate view therefore makes using pan-privacy easier.

We first analyze the denominator of $\frac{\mathbb{P}[M_{t-1}=m^* | Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]}$. We can rewrite it as

$$\mathbb{P}\left[M_{t-1} = m^*\right] = \sum_{q=0}^{t-1} \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = q\right] \cdot \mathbb{P}\left[N^\alpha_{j,t-1} = q\right]. \tag{10}$$

Fix some $q \in \{0, 1, \ldots, t-1\}$. Let $S_{j,\leq t^*}$ be the random variable for the bin pairs and component of j sampled through time t^*, i.e. $S_{j,\leq t^*} = \{(J_t, E^\alpha_{j,t})\}_{t=1}^{t^*}$. Note that this means the tuple $(j', 1)$ is possible only when $j' = j$. Define $\mathcal{S}^\alpha_{j,q,t}$ to be the set of realizations of $S_{j,\leq t}$ with exactly q samples from the α component of bin pair j. Then

$$\begin{aligned}\mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = q\right] &= \sum_{s \in \mathcal{S}^\alpha_{j,q,t-1}} \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s\right] \cdot \mathbb{P}\left[S_{j,\leq t-1} = s \mid N^\alpha_{j,t-1} = q\right] \\ &= \sum_{s \in \mathcal{S}^\alpha_{j,q,t-1}} \frac{1}{\binom{t-1}{q} k^{t-1-q}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s\right] \end{aligned} \tag{11}$$

where the second equality uses the fact that, conditioned on $N^\alpha_{j,t-1} = q$, there are $\binom{t-1}{q} k^{t-1-q}$ equiprobable realizations of $S_{j,\leq t-1}$. Note that we are now reasoning directly about the stream's effect on the state M_{t-1}. This is much closer to the application of pan-privacy that we set out to achieve.

Consider a length-$(t-1)$ realization $s \in \mathcal{S}^\alpha_{j,q,t-1}$. Recall that each index of s takes one of $j+1$ possible values: $(1, 0), (2, 0), \ldots, (k, 0)$, or $(j, 1)$. Let $s' \in S^\alpha_{j,0,t-1}$ be a realization such that the Hamming distance $d_H(s, s') = q$, i.e. s and s' differ in exactly q indices. Then because M_{t-1} is an ε-differentially private function of the stream, by group privacy (see e.g. Theorem 2.2 in Dwork et al. (2014))

$$\mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s\right] \leq e^{q\varepsilon} \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s'\right].$$

Moreover, there are exactly k^q such s' for each such s. Denote this set of s' by $T_{s,q}$. We can now continue

$$\begin{aligned}
(11) &= \sum_{s \in \mathcal{S}^\alpha_{j,q,t-1}} \frac{1}{k^q} \sum_{s' \in T_{s,q}} \frac{1}{\binom{t-1}{q} k^{t-1-q}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s\right] \\
&\leq \sum_{s \in \mathcal{S}^\alpha_{j,q,t-1}} \sum_{s' \in T_{s,q}} \frac{e^{q\varepsilon}}{\binom{t-1}{q} k^{t-1}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s'\right] \\
&= \sum_{s' \in S^\alpha_{j,0,t-1}} \frac{e^{q\varepsilon}}{k^{t-1}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s'\right] \\
&= \sum_{s' \in S^\alpha_{j,0,t-1}} e^{q\varepsilon} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s'\right] \cdot \mathbb{P}\left[S_{j,\leq t-1} = s' \mid N^\alpha_{j,t-1} = 0\right] \\
&= e^{q\varepsilon} \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right]
\end{aligned}$$

where the first inequality uses the above group privacy guarantee; the second equality uses the fact that, for a given $s' \in T_{s,q}$, there are exactly $\binom{t-1}{q}$ length-$(t-1)$ realizations s with q samples from the α mixture component from bin pair j and $d_H(s, s') = q$; and the last equality uses the fact that M_{t-1} and $N^\alpha_{j,t-1}$ are independent conditioned on $S_{j,\leq t-1}$. Note that this expression depending only on the conditioning for $N^\alpha_{j,t-1} = 0$ is useful because it will give us a "fixed point" to relate the numerator and denominator analyses. By expressing both quantities with respect to this condition, we can better compare them (and in particular, obtain a cancellation in the final ratio).

Returning to Equation 10

$$\mathbb{P}\left[M_{t-1} = m^*\right] = \sum_{q=0}^{t-1} \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = q\right] \cdot \mathbb{P}\left[N^\alpha_{j,t-1} = q\right]$$

we get

$$\begin{aligned}
\mathbb{P}\left[M_{t-1} = m^*\right] &\leq \sum_{q=0}^{t-1} e^{q\varepsilon} \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right] \cdot \mathbb{P}\left[N^\alpha_{j,t-1} = q\right] \\
&= \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right] \cdot \sum_{q=0}^{t-1} e^{q\varepsilon} \mathbb{P}\left[N^\alpha_{j,t-1} = q\right] \\
&= \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right] \cdot \mathbb{E}\left[e^{\varepsilon N^\alpha_{j,t-1}}\right]. \qquad (12)
\end{aligned}$$

To analyze this last quantity, recall that we defined random variable $E^\alpha_{j,t}$ as the indicator variable for drawing stream element t from the α mixture component of bin pair j. Then

$$\mathbb{E}\left[e^{\varepsilon N_{j,t-1}}\right] = \mathbb{E}\left[e^{\sum_{i=1}^{t-1} \varepsilon E^\alpha_{j,i}}\right] = \prod_{i=1}^{t-1} \mathbb{E}\left[e^{\varepsilon E^\alpha_{j,i}}\right] = \left[\left(1 - \frac{\alpha}{k}\right) e^0 + \frac{\alpha}{k} e^\varepsilon\right]^{t-1} = \left[1 + \frac{\alpha(e^\varepsilon - 1)}{k}\right]^{t-1}.$$

Since $1 + x \leq e^x$, $[1 + \frac{\alpha(e^\varepsilon - 1)}{k}]^{t-1} \leq e^{\frac{\alpha(e^\varepsilon - 1)(t-1)}{k}}$. We analyze this quantity in cases.

In the first case, $\frac{\alpha(e^\varepsilon-1)(t-1)}{k} \geq 1$. Then $t > \frac{k}{\alpha(e^\varepsilon-1)}$, and since $\varepsilon = O(1)$ there exists constant C such that $t > C\frac{k}{\alpha\varepsilon}$. $t \leq m$ so $m > C\frac{k}{\alpha\varepsilon}$. However, by the non-private uniformity testing lower bound, $I(X; M_m) = \Omega(1)$ requires $m = \Omega\left(\frac{\sqrt{k}}{\alpha^2}\right)$. This means we have some constant C' such that

$$m > C'\left(\frac{\sqrt{k}}{\alpha^2}\right)^{1/3}\left(\frac{k}{\alpha\varepsilon}\right)^{2/3} = \Omega\left(\tfrac{k^{5/6}}{\alpha^{4/3}\varepsilon^{2/3}}\right) \tag{13}$$

which suffices for our overall lower bound.

All that remains is the second case, $\frac{\alpha(e^\varepsilon-1)(t-1)}{k} < 1$. Then since $e^x \leq 1 + 2x$ for $x \in [0, 1]$, $e^{\frac{\alpha(e^\varepsilon-1)(t-1)}{k}} \leq 1 + 2\frac{\alpha(e^\varepsilon-1)(t-1)}{k}$. Again using $\varepsilon = O(1)$, there exists constant C_1 such that $\left[1 + \frac{\alpha(e^\varepsilon-1)}{k}\right]^{t-1} \leq e^{\frac{\alpha(e^\varepsilon-1)(t-1)}{k}} \leq 1 + C_1\frac{\alpha\varepsilon(t-1)}{k}$. Thus we return to Equation 12 and get

$$\mathbb{P}\left[M_{t-1} = m^*\right] \leq \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right] \cdot \left(1 + C_1\tfrac{\alpha\varepsilon(t-1)}{k}\right).$$

If we repeat this process using the other direction of group privacy, we get

$$\mathbb{P}\left[M_{t-1} = m^*\right] \geq \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right]\left[1 + \frac{\alpha(e^{-\varepsilon}-1)}{k}\right]^{t-1}.$$

$k \geq 2$, $\varepsilon > 0$, and $\alpha \leq 1$, so $\frac{\alpha(e^{-\varepsilon}-1)}{k} \in (-1, 0)$. Thus $\left[1 + \frac{\alpha(e^{-\varepsilon}-1)}{k}\right]^{t-1} \geq 1 + \frac{\alpha(e^{-\varepsilon}-1)(t-1)}{k}$. By $\varepsilon = O(1)$, we get a constant C_2 such that $\left[1 + \frac{\alpha(e^{-\varepsilon}-1)}{k}\right]^{t-1} \geq 1 - C_2\frac{\alpha\varepsilon(t-1)}{k}$. Tracing back,

$$\mathbb{P}\left[M_{t-1} = m^*\right] \geq \mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = 0\right] \cdot \left(1 - C_2\tfrac{\alpha\varepsilon(t-1)}{k}\right).$$

Returning to the beginning of our proof, we can repeat the argument for the numerator of $\frac{\mathbb{P}[M_{t-1}=m^* \mid Y_j=1]}{\mathbb{P}[M_{t-1}=m^*]}$:

$$\begin{aligned}\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right] &= \sum_{q=0}^{t-1}\mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = q, Y_j = 1\right] \cdot \mathbb{P}\left[N^\alpha_{j,t-1} = q \mid Y_j = 1\right] \\ &= \sum_{q=0}^{t-1}\mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = q, Y_j = 1\right] \cdot \mathbb{P}\left[N^\alpha_{j,t-1} = q\right]\end{aligned}$$

since $N^\alpha_{j,t}$ and Y_j are independent. Fixing a q, we rewrite $\mathbb{P}\left[M_{t-1} = m^* \mid N^\alpha_{j,t-1} = q, Y_j = 1\right]$

$$\begin{aligned}&= \sum_{s\in\mathcal{S}^\alpha_{j,q,t-1}}\mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s, Y_j = 1\right] \cdot \mathbb{P}\left[S_{j,\leq t-1} = s \mid N^\alpha_{j,t-1} = q\right] \\ &= \sum_{s\in\mathcal{S}^\alpha_{j,q,t-1}}\frac{1}{\binom{t-1}{q}k^{t-1-q}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s, Y_j = 1\right] \end{aligned} \tag{14}$$

where the first equality uses the independence of M_{t-1} and $N^{\alpha}_{j,t-1}$ conditioned on $S_{j,t-1}$ as well as the independence of $S_{j,\leq t-1}$ and Y_j, and the second equality uses the same counting argument as in the denominator case. Next, ε-pan-privacy gives

$$\mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s, Y_j = 1\right] \leq e^{q\varepsilon}\mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s', Y_j = 1\right]$$

and so

$$\begin{aligned}
(14) &= \sum_{s \in \mathcal{S}^{\alpha}_{j,q,t-1}} \frac{1}{k^q} \sum_{s' \in T_{s,q}} \frac{1}{\binom{t-1}{q} k^{t-1-q}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,t-1} = s, Y_j = 1\right] \\
&\leq \sum_{s \in \mathcal{S}^{\alpha}_{j,q,t-1}} \sum_{s' \in T_{s,q}} \frac{e^{q\varepsilon}}{\binom{t-1}{q} k^{t-1}} \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,t-1} = s', Y_j = 1\right] \\
&= \sum_{s' \in \mathcal{S}^{\alpha}_{j,q,t-1}} \frac{e^{q\varepsilon}}{k^{t-1}} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,t,-1} = s', Y_j = 1\right] \\
&= \sum_{s' \in \mathcal{S}^{\alpha}_{j,0,t-1}} e^{q\varepsilon} \cdot \mathbb{P}\left[M_{t-1} = m^* \mid S_{j,\leq t-1} = s', Y_j = 1\right] \cdot \mathbb{P}\left[S_{j,\leq t-1} = s' \mid N^{\alpha}_{j,t-1} = 0, Y_j = 1\right] \\
&= e^{q\varepsilon}\mathbb{P}\left[M_{t-1} = m^* \mid N^{\alpha}_{j,t-1} = 0\right]
\end{aligned}$$

where the last equality uses the independence of $S_{j,\leq t-1}$ and Y_j conditioned on $N^{\alpha}_{j,t-1} = 0$ and the independence of M_{t-1} and Y_j and $N^{\alpha}_{j,t-1}$ conditioned on $S_{j,\leq t-1}$. In turn we get

$$\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right] \leq \mathbb{P}\left[M_{t-1} = m^* \mid N^{\alpha}_{j,t-1} = 0\right] \sum_{q=0}^{t-1} e^{q\varepsilon} \mathbb{P}\left[N^{\alpha}_{j,t-1} = q\right]$$

which is the same quantity as in Equation 12. The same analysis thus gives

$$\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right] \leq \mathbb{P}\left[M_{t-1} = m^* \mid N^{\alpha}_{j,t-1} = 0\right] \cdot \left(1 + C_1 \frac{\alpha\varepsilon(t-1)}{k}\right)$$

as in the denominator case, and

$$\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right] \geq \mathbb{P}\left[M_{t-1} = m^* \mid N^{\alpha}_{j,t-1} = 0\right] \cdot \left(1 - C_2 \frac{\alpha\varepsilon(t-1)}{k}\right).$$

Summing up, both $\mathbb{P}\left[M_{t-1} = m^*\right]$ and $\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right]$ lie in the interval

$$\left[\mathbb{P}\left[M_{t-1} = m^* \mid N^{\alpha}_{j,t-1} = 0\right] \cdot \left(1 - C_2 \frac{\alpha\varepsilon(t-1)}{k}\right), \mathbb{P}\left[M_{t-1} = m^* \mid N^{\alpha}_{j,t-1} = 0\right] \cdot \left(1 + C_1 \frac{\alpha\varepsilon(t-1)}{k}\right)\right].$$

Thus

$$\begin{aligned}
\frac{\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right]}{\mathbb{P}\left[M_{t-1} = m^*\right]} &\leq \frac{1 + C_1 \frac{\alpha\varepsilon(t-1)}{k}}{1 - C_2 \frac{\alpha\varepsilon(t-1)}{k}} \\
&= 1 + \frac{C_1 + C_2}{1 - C_2 \frac{\alpha\varepsilon(t-1)}{k}} \cdot \frac{\alpha\varepsilon(t-1)}{k} \\
&= 1 + O\left(\frac{\alpha\varepsilon t}{k}\right)
\end{aligned}$$

where the last equality uses $\frac{\alpha\varepsilon(t-1)}{k} < \frac{1}{2C_2}$ (otherwise, we get $m = \Omega\left(\frac{k}{\alpha\varepsilon}\right)$ and can use the argument given in Equation 13). Similarly,

$$\begin{aligned}\frac{\mathbb{P}\left[M_{t-1} = m^* \mid Y_j = 1\right]}{\mathbb{P}\left[M_{t-1} = m^*\right]} &\geq \frac{1 - C_2\frac{\alpha\varepsilon(t-1)}{k}}{1 + C_1\frac{\alpha\varepsilon(t-1)}{k}} \\ &= 1 - \frac{C_1 + C_2}{1 + C_1\frac{\alpha\varepsilon(t-1)}{k}} \cdot \frac{\alpha\varepsilon(t-1)}{k} \\ &= 1 - O\left(\frac{\alpha\varepsilon t}{k}\right)\end{aligned}$$

and the claim follows. ■

Lemma 24 gives $A \leq \frac{\alpha^2\varepsilon^2 t^2}{k}$, so $\frac{\alpha^2 A}{k} \leq \frac{\alpha^4\varepsilon^2 t^2}{k^2}$. Returning to Equation 9 and using $t \leq m$, $I(X; V_t \mid M_{t-1}, J_t) = O\left(\frac{\alpha^4\varepsilon^2 m^2}{k^2}\right)$. Then we trace back to Equation 6 and get $I(X; M_m) = O\left(\frac{\alpha^4\varepsilon^2 m^3}{k^2}\right)$. Finally, a uniformity tester requires $I(X; M_m) = \Omega(1)$, so $m = \Omega\left(\frac{k^{2/3}}{\alpha^{4/3}\varepsilon^{2/3}}\right)$. ■

Theorem 25 *For $\varepsilon = O(1)$, any sequentially interactive ε-locally private uniformity tester requires $m = \Omega\left(\frac{k}{\alpha^2\varepsilon^2}\right)$ samples.*

Proof Let M_t be the random variable for the message sent by user t with sample S_t, and let $M_{1:t}$ be the concatenation of messages sent through time t. We start by distinguishing our approach for this lower bound from its pan-private analogue. Recall that in the pan-private lower bound we expressed the mutual information between the distribution parameter X and the internal state after m samples M_m as $I(X; M_m) = \sum_{t=1}^m I(X; S_t \mid M_{t-1})$. Here, we want to control the mutual information between X and the transcript through m samples, $I(X; M_{1:m})$. A key difference in the local setting is that the algorithm does not see any sample S_t. Instead, the algorithm sees a randomizer output based on S_t. We should therefore expect some information loss between the sample and its randomizer output. We formalize this using existing local privacy work (Lemma 26) and get $I(X; M_{1:m}) < \sum_{t=1}^m O(\varepsilon^2) \cdot I(X; S_t \mid M_{1:t-1})$. This partially explains the locally private lower bound's different dependence on ε.

More formally, by the chain rule for mutual information, $I(X; M_{1:m}) = \sum_{t=1}^m I(X; M_t \mid M_{1:t-1})$. Choose one term $I(X; M_t \mid M_{1:t-1})$ and fix a value m for $M_{1:t-1}$. By the KL divergence formulation of mutual information (see e.g. Fact 2) we can rewrite $I(X; M_t \mid M_{1:t-1} = m)$ as

$$\begin{aligned}&= \mathbb{E}_{X \mid M_{1:t-1} = m}\left[D_{KL}\left(M_t \mid X, M_{1:t-1} = m || M_t \mid M_{1:t-1} = m\right)\right] \\ &= \mathbb{P}\left[X = 0 \mid M_{1:t-1} = m\right] D_{KL}\left(M_t \mid X = 0, M_{1:t-1} = m || M_t \mid M_{1:t-1} = m\right) \\ &\quad + \mathbb{P}\left[X = 1 \mid M_{1:t-1} = m\right] D_{KL}\left(M_t \mid X = 1, M_{1:t-1} = m || M_t \mid M_{1:t-1} = m\right). \qquad (15)\end{aligned}$$

$M_{1:m}$ is generated by a sequentially interactive ε-locally private protocol. We can therefore use the following result from Duchi et al. (2013).

Lemma 26 (Theorem 1 (Duchi et al. (2013))) *Let Q be the output distribution for an ε-local randomizer in a sequentially interactive protocol. For any two input distributions P_1 and P_2, the induced output distributions Q_1 and Q_2 have*

$$D_{KL}\left(Q_1 || Q_2\right) + D_{KL}\left(Q_2 || Q_1\right) \leq 4(e^\varepsilon - 1)^2 || P_1 - P_2 ||_{TV}^2.$$

Here, we let P_1 be the distribution for $S_t \mid M_{1:t-1} = m$, P_2 for $S_t \mid X = 0, M_{1:t-1} = m$, and P_3 for $S_t \mid X = 1, M_{1:t-1} = m$. Q_1 is then the distribution for $M_t \mid M_{1:t-1} = m$, Q_2 for $M_t \mid X = 0, M_{1:t-1} = m$, and Q_3 for $M_t \mid X = 1, M_{1:t-1} = m$. Lemma 26 then gives

$$\begin{aligned}(15) &\leq 4(e^\varepsilon - 1)^2 \left[\mathbb{P}\left[X = 0 \mid M_{1:t-1} = m\right] ||P_1 - P_2||_{TV}^2 + \mathbb{P}\left[X = 1 \mid M_{1:t-1} = m\right] ||P_1 - P_3||_{TV}^2\right] \\ &\leq 2(e^\varepsilon - 1)^2 [\mathbb{P}\left[X = 0 \mid M_{1:t-1} = m\right] D_{KL}\left(P_1 || P_2\right) + \mathbb{P}\left[X = 1 \mid M_{1:t-1} = m\right] D_{KL}\left(P_1 || P_3\right)] \\ &= 2(e^\varepsilon - 1)^2 I(X; S_t \mid M_{1:t-1} = m)\end{aligned}$$

where the second inequality uses Pinsker's inequality (Lemma 31 in Appendix 11). Now we can quantify the loss in information between the sample S_t and the private message M_t:

$$\begin{aligned} I(X; M_{1:m}) &= \sum_{t=1}^{m} I(X; M_t \mid M_{1:t-1}) \\ &\leq \sum_{t=1}^{m} 2(e^\varepsilon - 1)^2 I(X; S_t \mid M_{1:t-1}) \\ &\leq \sum_{t=1}^{m} 2(e^\varepsilon - 1)^2 I(X; V_t \mid M_{1:t-1}, J_t) \end{aligned} \tag{16}$$

and, by the same reasoning as in the proof of Theorem 8,

$$I(X; V_t \mid M_{1:t-1}, J_t) = O\left(\frac{\alpha^2}{k} \mathbb{E}_{M_{1:t-1}}\left[\sum_{j=1}^{k} \mathbb{E}\left[Y_j \mid M_{1:t-1}\right]^2\right]\right). \tag{17}$$

and in turn rewrite the RHS inside $O\left(\cdot\right)$ as

$$\frac{\alpha^2}{k} \sum_{i=1}^{t-1} \sum_{j=1}^{k} \left(\mathbb{E}_{M_{1:i}}\left[\mathbb{E}\left[Y_j \mid M_{1:i}\right]^2\right] - \mathbb{E}_{M_{1:i-1}}\left[\mathbb{E}\left[Y_j \mid M_{1:i-1}\right]^2\right]\right). \tag{18}$$

We now fix some i and want to upper bound

$$\sum_{j=1}^{k} \left(\mathbb{E}_{M_{1:i}}\left[\mathbb{E}\left[Y_j \mid M_{1:i}\right]^2\right] - \mathbb{E}_{M_{1:i-1}}\left[\mathbb{E}\left[Y_j \mid M_{1:i-1}\right]^2\right]\right).$$

Choose one term j and define $\gamma_j = \mathbb{P}\left[Y_j = 1 \mid M_{1:i}\right]$. Then we get

$$\begin{aligned}\mathbb{E}_{M_{1:i}}\left[\mathbb{E}\left[Y_j \mid M_{1:i}\right]^2\right] &= \mathbb{E}_{M_{1:i}}\left[(\gamma_j - (1 - \gamma_j))^2\right] \\ &= \mathbb{E}_{M_{1:i}}\left[4\gamma_j^2 - 4\gamma_j + 1\right] \\ &= 4\mathbb{E}_{M_{1:i}}\left[\gamma_j^2\right] - 4\mathbb{E}_{M_{1:i}}\left[\gamma_j\right] + 1 \\ &= 4\mathbb{E}_{M_{1:i}}\left[\gamma_j^2\right] - 1\end{aligned}$$

where the last equality uses $4\mathbb{E}_{M_{1:i}}\left[\gamma_j\right] = 4\mathbb{E}_{M_{1:i}}\left[\mathbb{P}\left[Y_j = 1 \mid M_{1:i}\right]\right] = 4\mathbb{P}\left[Y_j = 1\right] = 2$. By similar reasoning, if we define $\eta_j = \mathbb{P}\left[Y_j = 1 \mid M_{1:i-1}\right]$ then we get

$$\mathbb{E}_{M_{1:i-1}}\left[\mathbb{E}\left[Y_j \mid M_{1:i-1}\right]^2\right] = 4\mathbb{E}_{M_{1:i-1}}\left[\eta_j^2\right] - 1.$$

Tracing back, our goal is now to upper bound

$$\mathbb{E}_{M_{1:i}}\left[\mathbb{E}\left[Y_j \mid M_{1:i}\right]^2\right] - \mathbb{E}_{M_{1:t-1}}\left[\mathbb{E}\left[Y_j \mid M_{1:i-1}\right]^2\right] = 4\left(\mathbb{E}_{M_{1:i}}\left[\gamma_j^2\right] - \mathbb{E}_{M_{1:i-1}}\left[\eta_j^2\right]\right). \quad (19)$$

Our analysis will be easier if we restrict the message space for $M_1, \ldots, M_i$ to be binary. We do so by a result from Bassily and Smith (2015). This again relies on the local privacy of the protocol.

Lemma 27 (Theorem 4.1 in Bassily and Smith (2015)) *Given a sequentially interactive ε-locally private protocol with expected number of randomizer calls T, there exists an equivalent sequentially interactive ε-locally private protocol with expected sample complexity $e^\varepsilon T$ where each user sends a single bit from a single randomizer call.*

The cost of this transformation is an e^ε blowup in expected sample complexity and an additional $O(m\log(\log(m)))$ bits of public randomness. First, since we assumed $\varepsilon = O(1)$, by Markov's inequality we can trade an arbitrarily small constant c decrease in overall success probability for a constant ($O(e^\varepsilon/c) = O(1)$) blowup in sample complexity. Combined with our assumption of arbitrary access to public randomness for locally private protocols, it is without loss of generality to assume all of our $M_1, \ldots, M_i$ are binary.[5]

Returning to $4\left(\mathbb{E}_{M_{1:i}}\left[\gamma_j^2\right] - \mathbb{E}_{M_{1:i-1}}\left[\eta_j^2\right]\right)$ in Equation (19), suppose we fix $M_{1:i-1}$ below. Then

$$\begin{aligned}
\mathbb{E}_{M_{1:i}}\left[\gamma_j^2\right] &= \mathbb{P}\left[M_i = 1\right] \cdot \mathbb{P}\left[Y_j = 1 \mid M_i = 1\right]^2 + \mathbb{P}\left[M_i = 0\right] \cdot \mathbb{P}\left[Y_j = 1 \mid M_i = 0\right]^2 \\
&= \frac{\left[\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right] \cdot \mathbb{P}\left[Y_j = 1\right]\right]^2}{\mathbb{P}\left[M_i = 1\right]} + \frac{\left[\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right] \cdot \mathbb{P}\left[Y_j = 1\right]\right]^2}{\mathbb{P}\left[M_i = 0\right]} \\
&= \eta_j^2\left[\frac{\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]^2}{\mathbb{P}\left[M_i = 1\right]} + \frac{\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right]^2}{\mathbb{P}\left[M_i = 0\right]}\right]
\end{aligned}$$

where the second equality uses Bayes' rule. Now, using $-2x + 2y - 2(1-x) + 2(1-y) = 0$ with $x = \mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]$ and $y = \mathbb{P}\left[M_i = 1\right]$, we get

$$-2\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right] + 2\mathbb{P}\left[M_i = 1\right] - 2\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right] + 2\mathbb{P}\left[M_j = 0\right] = 0.$$

We can now add 0 inside the bracketed term to get

$$\eta_j^2\left[\frac{\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]^2}{\mathbb{P}\left[M_i = 1\right]} + \frac{\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right]^2}{\mathbb{P}\left[M_i = 0\right]}\right] = \eta_j^2\left[A + B\right]$$

where

$$\begin{aligned}
A &= \frac{\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]^2 - 2\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]\mathbb{P}\left[M_i = 1\right] + 2\mathbb{P}\left[M_i = 1\right]^2}{\mathbb{P}\left[M_i = 1\right]} \\
&= \frac{\left(\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right] - \mathbb{P}\left[M_i = 1\right]\right)^2}{\mathbb{P}\left[M_i = 1\right]} + \mathbb{P}\left[M_i = 1\right]
\end{aligned}$$

[5] Note that this step relies on the fact that, in sequentially interactive protocols, the number of randomizer calls is the same as the sample complexity. For fully interactive protocols, the number of randomizer calls may arbitrarily exceed the sample complexity. However, using the transformation given by Joseph et al. (2019), our argument also extends to any $O(1)$-compositional fully interactive protocol.

and

$$\begin{aligned} B &= \frac{\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right]^2 - 2\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right]\mathbb{P}\left[M_i = 0\right] + 2\mathbb{P}\left[M_i = 0\right]^2}{\mathbb{P}\left[M_i = 0\right]} \\ &= \frac{\left(\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right] - \mathbb{P}\left[M_i = 0\right]\right)^2}{\mathbb{P}\left[M_i = 0\right]} + \mathbb{P}\left[M_i = 0\right]. \end{aligned}$$

Thus we may rewrite $\eta_j^2[A + B]$ as

$$\eta_j^2 \left[1 + \frac{\left(\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right] - \mathbb{P}\left[M_i = 1\right]\right)^2}{\mathbb{P}\left[M_i = 1\right]} + \frac{\left(\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right] - \mathbb{P}\left[M_i = 0\right]\right)^2}{\mathbb{P}\left[M_i = 0\right]}\right]. \tag{20}$$

For neatness, let $C = \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i = j\right]$ and $D = \mathbb{P}\left[M_i = 1 \mid Y_j = -1, J_i = j\right]$. Recall that J_i denotes which of k bin pairs is chosen. Then

$$\begin{aligned} \mathbb{P}\left[M_i = 1 \mid Y_j = 1\right] &= \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i \neq j\right] \cdot \mathbb{P}\left[J_i \neq j \mid Y_j = 1\right] \\ &\quad + \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i = j\right] \cdot \mathbb{P}\left[J_i = j \mid Y_j = 1\right] \\ &= \frac{k-1}{k} \cdot \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i \neq j\right] + \frac{C}{k} \end{aligned}$$

since J_i is independent of Y_j and $\mathbb{P}\left[J_i = j\right] = \frac{1}{k}$. Similarly,

$$\begin{aligned} \mathbb{P}\left[M_i = 1\right] &= \mathbb{P}\left[M_i = 1 \mid J_i \neq j\right] \cdot \mathbb{P}\left[J_i \neq j\right] + \mathbb{P}\left[M_i = 1 \mid J_i = j\right] \cdot \mathbb{P}\left[J_i = j\right] \\ &= \frac{k-1}{k} \cdot \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i \neq j\right] \\ &\quad + \frac{1}{k} \cdot \left(\mathbb{P}\left[M_i = 1 \mid J_i = j, Y_j = 1\right] \cdot \mathbb{P}\left[Y_j = 1\right] + \mathbb{P}\left[M_i = 1 \mid J_i = j, Y_j = -1\right] \cdot \mathbb{P}\left[Y_j = -1\right]\right) \\ &= \frac{k-1}{k} \cdot \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i \neq j\right] + \frac{1}{k}\left(\eta_j C + (1 - \eta_j) D\right) \end{aligned}$$

where the second equality uses the conditional independence of M_i and Y_j given $J_t \neq j$ and fixed $M_{1:i-1}$. We substitute these expressions for $\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]$ and $\mathbb{P}\left[M_i = 1\right]$ and get

$$\left(\mathbb{P}\left[M_i = 1 \mid Y_j = 1\right] - \mathbb{P}\left[M_i = 1\right]\right)^2 = \left[\frac{(1 - \eta_j)(C - D)}{k}\right]^2 = \left(\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right] - \mathbb{P}\left[M_i = 0\right]\right)^2$$

where the last equality follows from $\mathbb{P}\left[M_i = 0 \mid Y_j = 1\right] = 1 - \mathbb{P}\left[M_i = 1 \mid Y_j = 1\right]$ and $\mathbb{P}\left[M_i = 1\right] = 1 - \mathbb{P}\left[M_i = 0\right]$. Returning to Equation (20), we have

$$\begin{aligned} \eta_j^2\left[A + B\right] &= \eta_j^2\left[1 + \left(\frac{(1 - \eta_i)(C - D)}{k}\right)^2 \left(\frac{1}{\mathbb{P}\left[M_i = 1\right]} + \frac{1}{\mathbb{P}\left[M_i = 0\right]}\right)\right] \\ &= \eta_j^2\left[1 + \left(\frac{(1 - \eta_i)(C - D)}{k}\right)^2 \cdot \frac{1}{\mathbb{P}\left[M_i = 1\right]\mathbb{P}\left[M_i = 0\right]}\right] \end{aligned} \tag{21}$$

since $\mathbb{P}\left[M_i = 1\right] + \mathbb{P}\left[M_i = 0\right] = 1$. We now analyze $\frac{|C-D|}{\mathbb{P}[M_i=1]}$. It will be useful to recall the sampling thought experiment used in the proof of Lemma 24: at each time t, we first uniformly sample bin pair $J_t \sim_U [k]$ and then sample the bin from a mixture: having sampled bin pair j, with probability $1-\alpha$ we take a uniform random draw from $\{2j-1, 2j\}$. With the remaining probability α, if $Y_j = 1$ then we sample $2j-1$, and if $Y_j = -1$ then we sample $2j$. Finally, we define $E^{\alpha}_{j,t} = 1$ if $J_t = j$ and we sample from the α mixture component and $E^{\alpha}_{j,t} = 0$ otherwise.

Under this equivalent sampling method, we can rewrite

$$\begin{aligned}
C &= \mathbb{P}\left[M_i = 1 \mid Y_j = 1, J_i = j\right] \\
&= \mathbb{P}\left[M_i = 1 \mid E^{\alpha}_{j,i} = 1, Y_j = 1, J_i = j\right] \mathbb{P}\left[E^{\alpha}_{j,i} = 1 \mid Y_j = 1, J_i = j\right] \\
&\quad + \mathbb{P}\left[M_i = 1 \mid E^{\alpha}_{j,i} = 0, Y_j = 1, J_i = j\right] \mathbb{P}\left[E^{\alpha}_{j,i} = 0 \mid Y_j = 1, J_i = j\right] \\
&= \alpha \mathbb{P}\left[M_i = 1 \mid Y_j = 1, E^{\alpha}_{j,i} = 1\right] + (1-\alpha)\mathbb{P}\left[M_i = 1 \mid E^{\alpha}_{j,i} = 0, J_i = j\right]
\end{aligned}$$

where the last equality uses the fact that M_i is independent of J_i conditioned on $E^{\alpha}_{j,i} = 1$ and M_i is independent of Y_j conditioned on $E^{\alpha}_{j,i} = 0$. Similarly

$$D = \alpha \mathbb{P}\left[M_i = 1 \mid Y_j = -1, E^{\alpha}_{j,i} = 1\right] + (1-\alpha)\mathbb{P}\left[M_i = 1 \mid E^{\alpha}_{j,i} = 0, J_i = j\right].$$

Thus we can rewrite

$$\begin{aligned}
\frac{|C-D|}{\mathbb{P}\left[M_i = 1\right]} &= \frac{|\alpha(\mathbb{P}\left[M_i = 1 \mid Y_j = 1, E^{\alpha}_{j,i} = 1\right] - \mathbb{P}\left[M_i = 1 \mid Y_j = -1, E^{\alpha}_{j,i} = 1\right])|}{\mathbb{P}\left[M_i = 1\right]} \\
&\le \frac{|\alpha(e^{\varepsilon} - e^{-\varepsilon})\mathbb{P}\left[M_i = 1\right]|}{\mathbb{P}\left[M_i = 1\right]} \\
&= O(\alpha\varepsilon)
\end{aligned}$$

where the inequality uses the ε-local privacy of M_i (recalling that we have been conditioning on $M_{1:i-1}$), and the equality uses $\varepsilon = O(1)$. Similarly, we get

$$\begin{aligned}
1 - C &= \mathbb{P}\left[M_i = 0 \mid Y_j = 1, J_i = j\right] \\
&= \alpha \mathbb{P}\left[M_i = 0 \mid Y_j = 1, E^{\alpha}_{j,i} = 1\right] + (1-\alpha)\mathbb{P}\left[M_i = 0 \mid E^{\alpha}_{j,i} = 0, J_i = j\right]
\end{aligned}$$

and

$$1 - D = \alpha \mathbb{P}\left[M_i = 0 \mid Y_j = -1, E^{\alpha}_{j,i} = 1\right] + (1-\alpha)\mathbb{P}\left[M_i = 0 \mid E^{\alpha}_{j,i} = 0, J_i = j\right].$$

This gives us

$$\begin{aligned}
\frac{|C-D|}{\mathbb{P}\left[M_i = 0\right]} &= \frac{|(1-C) - (1-D)|}{\mathbb{P}\left[M_i = 0\right]} \\
&= \frac{|\alpha(\mathbb{P}\left[M_i = 0 \mid Y_j = 1, E^{\alpha}_{j,i} = 1\right] - \mathbb{P}\left[M_i = 0 \mid Y_j = -1, E^{\alpha}_{j,i} = 1\right])|}{\mathbb{P}\left[M_i = 1\right]} \\
&\le \frac{|\alpha(e^{\varepsilon} - e^{-\varepsilon})\mathbb{P}\left[M_i = 0\right]}{\mathbb{P}\left[M_i = 0\right]} \\
&= O(\alpha\varepsilon)
\end{aligned}$$

as well. Thus by Equation 21 $\eta_j^2[A+B] = \eta_j^2 + O\left(\frac{\eta_j^2(1-\eta_i)^2\alpha^2\varepsilon^2}{k^2}\right) = \eta_j^2 + O\left(\frac{\alpha^2\varepsilon^2}{k^2}\right)$ because $\eta_j^2(1-\eta_j)^2 < 1$. Returning to Equation (19), we can now bound

$$\mathbb{E}_{M_{1:i}}\left[\mathbb{E}\left[Y_j \mid M_{1:i}\right]^2\right] - \mathbb{E}_{M_{1:i-1}}\left[\mathbb{E}\left[Y_j \mid M_{1:i-1}\right]^2\right] = O\left(\frac{\alpha^2\varepsilon^2}{k^2}\right).$$

Since this analysis was for an arbitrary j, we get

$$\sum_{j=1}^{k}\left(\mathbb{E}_{M_{1:i}}\left[\mathbb{E}\left[Y_j \mid M_{1:i}\right]^2\right] - \mathbb{E}_{M_{1:i-1}}\left[\mathbb{E}\left[Y_j \mid M_{1:i-1}\right]^2\right]\right) = O\left(\frac{\alpha^2\varepsilon^2}{k}\right).$$

We substitute this into Equation (18) and get $I(X; V_t \mid M_{1:t-1}, J_t) = O\left(\frac{\alpha^4\varepsilon^2 t}{k^2}\right)$. Finally, substituting back into Equation (16) and using $t \leq m$ and $\varepsilon = O(1)$, $I(X; M_{1:m}) = O\left(\frac{\alpha^4\varepsilon^4 m^2}{k^2}\right)$. Since the output of a locally private algorithm is a function of the transcript, a uniformity tester with sample complexity m requires $I(X; M_{1:m}) = \Omega(1)$. We therefore get sample complexity $m = \Omega\left(\frac{k}{\alpha^2\varepsilon^2}\right)$. ■

11. Information Theory

Definition 28 *Let X be a random variable with probability mass function p_X. Then the* entropy *of X, denoted by $H(X)$, is defined as*

$$H(X) = \sum_x p_X(x)\log\left(\frac{1}{p_X(x)}\right),$$

and the conditional entropy *of random variable X conditioned on random variable Y is defined as $H(X|Y) = \mathbb{E}_y[H(X|Y=y)]$.*

Next, we can use entropy to define the mutual information between two random variables.

Definition 29 *The* mutual information *between two random variables X and Y is defined as $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$, and the* conditional mutual information *between X and Y given Z is defined as $I(X;Y|Z) = H(X|Z) - H(X|YZ) = H(Y|Z) - H(Y|XZ)$.*

Definition 30 *The* Kullback-Leibler divergence *between two random variables X and Y with probability mass functions p_X and p_Y is defined as*

$$D_{KL}\left(X||Y\right) = \sum_x p_X(x)\log\left(\frac{p_X(x)}{p_Y(x)}\right).$$

Fact 2 *Let X, Y, Z be random variables, we have*

$$I(X;Y|Z) = \mathbb{E}_{x,z}[D_{KL}\left((Y|X=x, Z=z)||(Y|Z=z)\right)].$$

Lemma 31 (Pinsker's inequality) *Let X and Y be random variables with probability mass functions p_X and p_Y. Then*

$$\sqrt{2D_{KL}\left(X||Y\right)} \geq 2||p_X - p_Y||_{TV}.$$