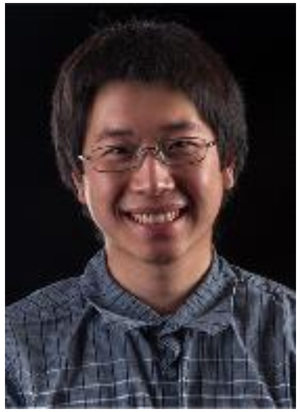
On Differential Privacy and Adaptive Data Analysis with Bounded Space

Itai Dinur

Uri Stemmer

David P. Woodruff

Samson Zhou



Streaming Model

- Input: Elements of an underlying data set S, which arrives sequentially
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size m of the input S

1 0 1 1 1 0 0 1

Frequency Vector

- Given a set S of m elements from $[n]$, let f_i be the frequency of element i. (How often it appears)

$$1\ 1\ 2\ 1\ 2\ 1\ 1\ 2\ 3 \rightarrow [5, 3, 1, 0] \coloneqq f$$

Frequency Moments (L_p Norm)

- Given a set S of m elements from $[n]$, let f_i be the frequency of element i. (How often it appears)
- Let F_p be the frequency moment of the vector:

$$F_p = f_1^p + f_2^p + \cdots + f_n^p$$

- Goal: Given a set S of m elements from $[n]$ and an accuracy parameter ε, output a $(1+\varepsilon)$-approximation to F_p
- Motivation: Entropy estimation, linear regression

Distinct Elements (F_0 Estimation)

- Given a set S of m elements from $[n]$, let f_i be the frequency of element i. (How often it appears)
- Let F_0 be the frequency moment of the vector:

$$F_0 = |\{i : f_i \neq 0\}|$$

- Goal: Given a set S of m elements from $[n]$ and an accuracy parameter ε, output a $(1+\varepsilon)$-approximation to F_0
- Motivation: Traffic monitoring

$(1+\varepsilon)$-Approximation Streaming Algorithms

- $O\left(\frac{\log n}{\varepsilon^2}\right)$ space streaming algorithm for F_2 estimation [AMS96]
  - Johnson-Lindenstrauss transformation [JL84]

- $O\left(\frac{1}{\varepsilon^2}+\log n\right)$ space streaming algorithm for F_2 estimation [KNW10]
  - Flajolet-Martin sketch [FM85]

Differential Privacy

- [DMNS06] Given $\varepsilon > 0$ and $\delta \in (0,1)$, a randomized algorithm $A: U^* \to Y$ is (ε, δ)-differentially private if, for every neighboring frequency vectors f and f' and for all $E \subseteq Y$,

$$\Pr[A(f) \in E] \leq e^{\varepsilon} \Pr[A(f') \in E] + \delta$$

$A($ $) \approx A($ $)$

$(1+\varepsilon)$-Approximation Streaming Algorithms

- $O\left(\frac{\log n}{\varepsilon^2}\right)$ space streaming algorithm for F_2 estimation [AMS96]
  - Johnson-Lindenstrauss transformation [JL84]
  - Johnson-Lindenstrauss transformation itself preserves DP [BBDS12]
- $O\left(\frac{1}{\varepsilon^2}+\log n\right)$ space streaming algorithm for F_2 estimation [KNW10]
  - Flajolet-Martin sketch [FM85]
  - Flajolet-Martin sketch itself preserves DP [SST20]

$(1 + \varepsilon)$-Approxim tion Str ning Algorithms

- $O\left(\frac{\log n}{\varepsilon^2}\right)$ spac [AMS96]
  - Johns
  - Johnson-
- $O\left(\frac{1}{\varepsilon^2} + \log n\right)$ sp alg r F_2 tion [KNW10]
  - Flajolet-Martin sketch [FM
  - Flajolet-Martin sketch itself reserves DP [SST20]

Does differential privacy cost more space?

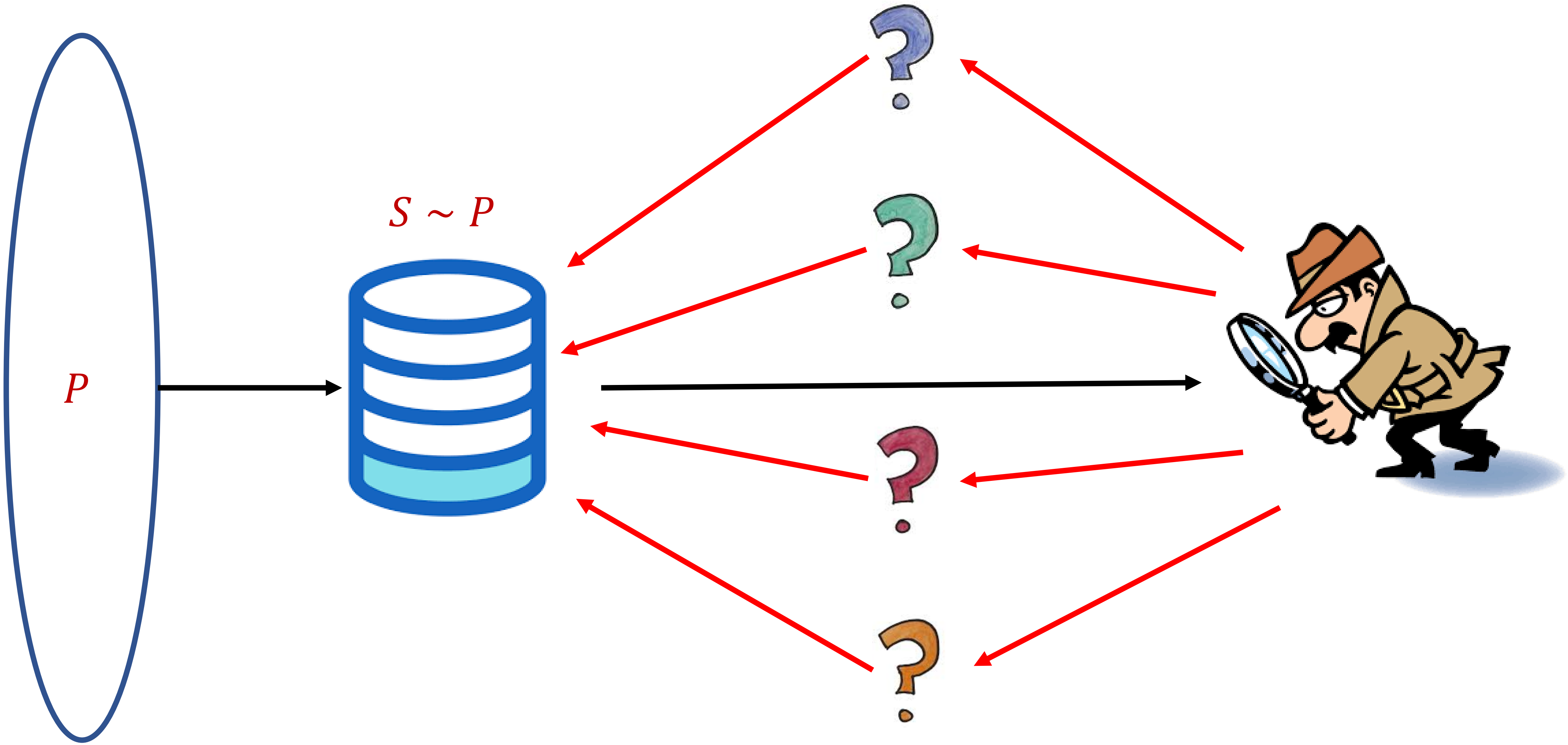
Our Results (Differential Privacy)

Let d be the “size” of the problem, i.e., data points from X can be represented using $\mathrm{polylog}(d)$ bits and queries from Q can be represented using $\mathrm{poly}(d)$ bits.

There exists a problem $P: X^* \times Q \rightarrow M$ such that:

1. P can be solved non-privately using $\mathrm{polylog}(d)$ bits of space
2. P can be solved privately using sample and space complexity $\tilde{O}(\sqrt{d})$
3. Any computationally-efficient differentially-private algorithm A for solving P must use space $\tilde{\Omega}(\sqrt{d})$ (assuming the existence of a sub-exponentially secure symmetric-key encryption scheme)

Adaptive Data Analysis



Adaptive Data Analysis

1. Adversary B chooses distribution P over a data domain X
2. Mechanism A obtains a sample $S \sim P^n$ containing n i.i.d. samples from P
3. For k rounds, $j = 1, \ldots, k$
    1. The adversary chooses a function $h_j: X \rightarrow \{-1,0,1\}$, possibly as a function of all previous answers given by the mechanism
    2. The mechanism obtains h_j and responds with an answer z_j, which is given to the adversary B

Adaptive Data Analysis

- Given n samples, there exists a computationally efficient oracle that accurately answers $\tilde{O}(n^2)$ adaptive queries [DFH+15]

- There is no computationally efficient oracle that given n samples is accurate on $\tilde{\Omega}(n^2)$ adaptively chosen queries (assuming the existence of one-way functions) [SU15]

Adaptive Data An lysis

- Given n … racle that accurately …
- There … les is accu …

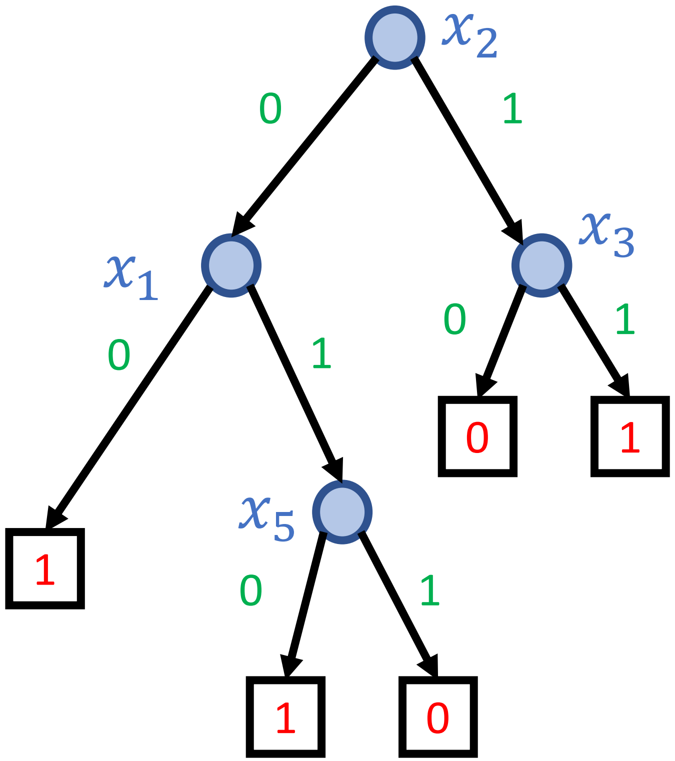
Is there a more fundamental bottleneck for the ADA problem than the number of samples?

Our Results (Adaptive Data Analysis)

Every computationally efficient mechanism that is $(0.1, 0.1)$-accurate for k queries must have space complexity at least $\Omega(\sqrt{k})$, assuming the existence of one-way functions

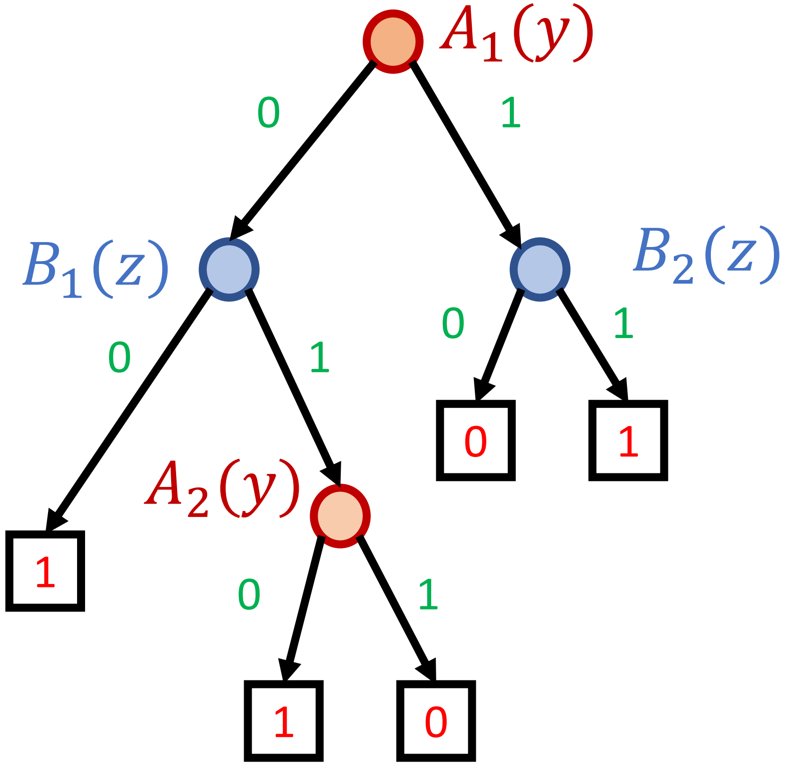
Query vs. Communication

$f(x)$



Decision Tree

$f(y \circ z)$



Communication Protocol

Talk Structure

- Multi-instance leakage-resilient (MILR) scheme definition
- Differential privacy separation
- Space bounded adaptive data analysis
- Construction of MILR

Questions?

Multi-Instance Leakage-Resilient Scheme

We define a multi-instance leakage-resilient scheme (or MILR scheme) to be a tuple of efficient algorithms (Gen, Param, Enc, Dec) :

- Gen is a randomized algorithm that takes as input a security parameter λ and outputs a λ-bit secret key, $x \leftarrow \mathrm{Gen}(1^{\lambda})$
- Param is a randomized algorithm that takes as input a security parameter λ and outputs a $\mathrm{poly}(\lambda)$-bit public parameter, $p \leftarrow \mathrm{Param}(1^{\lambda})$
- Enc is a randomized algorithm that takes as input a secret key x, a public parameter p, and a message $m \in \{0,1\}$ and outputs a ciphertext $\{0,1\}^{\mathrm{poly}(\lambda)}$, $c \leftarrow \mathrm{Enc}(x, p, m)$
- Dec is a deterministic algorithm that takes as input a secret key x, a public parameter p, and a ciphertext c, and outputs a decrypted message m', $m' \leftarrow \mathrm{Dec}(x, p, c)$. If $c = \mathrm{Enc}(x, p, m)$, then $m' = m$

Multi-Instance Leakage-Resilient Scheme

An MILR scheme (Gen, Param, Enc, Dec) is $(\Gamma, \overline{\tau})$-secure against space bounded pre-processing adversaries if both multi-semantic security and multi-security against bounded pre-processing adversary hold

Multi-Instance Leakage-Resilient Scheme

Let $\vec{x} = (x_1, \dots, x_n)$ be a vector of keys, and $\vec{p} = (p_1, \dots, p_n)$ be a vector of public parameters. Let $J \subseteq [n]$ be a set of "hidden coordinates". Define the two oracles:

1. $E_1(\vec{x}, \vec{p}, J, \cdot, \cdot)$ takes an index of a key $j \in [n]$ and a message m, and returns $\mathrm{Enc}(x_j, p_j, m)$
2. $E_0(\vec{x}, \vec{p}, J, \cdot, \cdot)$ takes an index of a key $j \in [n]$ and a message m. If $j \in J$, output $\mathrm{Enc}(x_j, p_j, 0)$. Otherwise if $j \notin J$, output $\mathrm{Enc}(x_j, p_j, m)$

Multi-Semantic Security

Given $\Gamma: \mathrm{R} \rightarrow \mathrm{R}$, every $n = \mathrm{poly}(\Gamma(\lambda))$ and every $\mathrm{poly}(\Gamma(\lambda))$-time adversary B, there exists negligible function negl

$$\left| \Pr_{\vec{x},\vec{p},B,\mathrm{Enc}}\left[B^{E_0(\vec{x},\vec{p},[n],\cdot,\cdot)}(\vec{p}) = 1\right] - \Pr_{\vec{x},\vec{p},B,\mathrm{Enc}}\left[B^{E_0(\vec{x},\vec{p},[n],\cdot,\cdot)}(\vec{p}) = 1\right] \right|$$

$$\leq \mathrm{negl}\,(\Gamma(\lambda))$$

“A computationally bounded adversary that gets the public parameters but not the keys, cannot tell whether it is interacting with E_0 or with E_1”

Multi-Security Against Bounded Pre-Processing Adversary

Given $\Gamma: \mathrm{R} \to \mathrm{R}$, every $n = \mathrm{poly}\big(\Gamma(\lambda)\big)$, pre-processing function F that outputs $z \leftarrow F(\vec{x})$ with $|z| \le s$, we can output a random $J \subseteq [n]$ with $|J| \ge n - \overline{\tau}(\lambda, s)$ such that for every $\mathrm{poly}(\Gamma(\lambda))$-time adversary B, there exists negligible function negl

$$\left| \Pr_{\vec{x},\vec{p},B,\mathrm{Enc},J,z}\left[B^{E_0(\vec{x},\vec{p},J,\cdot,\cdot)}(z,\vec{p}) = 1\right] - \Pr_{\vec{x},\vec{p},B,\mathrm{Enc},J,z}\left[B^{E_0(\vec{x},\vec{p},J,\cdot,\cdot)}(z,\vec{p}) = 1\right] \right|$$

$$\le \mathrm{negl}\,(\Gamma(\lambda))$$

"Even if s bits of our n keys are leaked then still encryptions w.r.t. the keys of J are computationally indistinguishable"

Multi-Instance Leakage-Resilient Scheme

An MILR scheme (Gen, Param, Enc, Dec) is $(\Gamma, \overline{\tau})$-secure against space bounded pre-processing adversaries if both multi-semantic security and multi-security against bounded pre-processing adversary hold

Theorem: If there exists a $\Gamma(\lambda)$-secure encryption scheme against non-uniform adversaries, then there exists an MILR scheme that is $(\Gamma(\lambda), \overline{\tau})$-secure against space bounded non-uniform preprocessing adversaries for $\overline{\tau} = \frac{2s}{\lambda} + 4$

Multi-Instance Leakage-Resilient Scheme

Intuition: Any good s-space-bounded adversary against an MILR can be viewed as a convex combination of adversaries that store $O\left(\frac{s}{\lambda}\right)$ samples

Talk Structure

- Multi-instance leakage-resilient (MILR) scheme definition
- Differential privacy separation
- Space bounded adaptive data analysis
- Construction of MILR

Space Hardness for Differential Privacy

Toy problem: Output either the last element of the stream or a $(1+\alpha)$-approximation to F_2

Non-private algorithm outputs the last element of the stream using $O(\log n)$ space

Private algorithm must output a $(1+\alpha)$-approximation to F_2, which requires $\Omega\left(\frac{1}{\alpha^2}\right)$ space [Woodruff04]

Space Hardness for Differential Privacy

Focus on the private and non-private algorithms computing “the same thing”

Consider algorithms that use a summary z of a dataset $D \in X^n$ to solve a problem $P: X^* \times Q \to M$, where Q is a family of possible queries, and M is a metric space

(α, β)-Accuracy

We say that $A = (A_1, A_2)$ solves a problem $P: X^* \times Q \to M$ with space complexity s, sample complexity n, error α, and confidence β if

- $A_1: X^* \to \{0,1\}^s$ is a pre-processing procedure that takes a dataset D and outputs an s bit string
- For every input dataset $D \in X^n$ and every query $q \in Q$ it holds that

$$\Pr_{\substack{\mathrm{z} \leftarrow A_1(D) \\ a \leftarrow A_2(z,q)}} [|a - P(D,q)| \leq \alpha] \leq \beta$$

Decrypted Average Vector (DAV)

Data set $D = (x_1, \ldots, x_n) \in \left(\{0,1\}^\lambda\right)^n$ of keys

Queries $q = ((p_1, c_1), \ldots, (p_n, c_n))$, public parameters p_i, ciphertexts c_i an encryption of a binary vector of length d

Output $\vec{a} = (a_1, \ldots, a_d) \in [0,1]^d$ to approximate (error in ℓ_∞)

$$\mathrm{dav}_q(D) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{Dec}(x_i, p_i, c_i)$$

Decrypted Average Vector (DAV)

Theorem: There exists a non-private streaming algorithm for the DAV problem with ℓ_∞ error $\frac{1}{10}$ that uses $O(\lambda \log d)$ bits of space

Algorithm: Sample $O(\log d)$ of the input keys, then estimate dav_q using the sampled keys for each query q

Decrypted Average Vector (DAV)

Theorem: There exists a (ε, δ)-private streaming algorithm for the DAV problem with ℓ_∞ error $\frac{1}{10}$ that uses $O\left(\frac{1}{\varepsilon}\sqrt{d \log \frac{1}{\delta}} \lambda \log d\right)$ bits of space

Algorithm: Sample $O\left(\frac{1}{\varepsilon}\sqrt{d \log \frac{1}{\delta}} \log d\right)$ the input keys, then estimate dav_q using the sampled keys for each query q with advanced composition [DRV10]

Decrypted Average Vector (DAV)

Theorem: Any computationally-efficient differentially-private algorithm A for solving the DAV problem with ℓ_∞ error $\frac{1}{10}$ must use space $\tilde{\Omega}(\sqrt{d})$ (assuming the existence of a sub-exponentially secure symmetric-key encryption scheme)

Theorem: Let Π be an MILR scheme that is $(\Gamma, \bar{\tau})$-secure against space bounded non-uniform preprocessing adversaries. For every $\mathrm{poly}(\Gamma(\lambda))$-time (ε, δ)-CDP algorithm for the DAV problem, we have $\bar{\tau} = \Omega\left(\sqrt{\frac{d}{\log n}}\right)$

Computational Differential Privacy

Given $\varepsilon > 0$ and $\delta \in (0,1)$, a randomized algorithm $A: U^* \to Y$ is (ε, δ)-computationally differentially private if, for neighboring datasets D and D' chosen by a $\mathrm{poly}(\lambda)$-time adversary (B, T), there exists a negligible function negl such that

$$\Pr_{(D_0,D_1)\leftarrow Q}\left[T\big(A(D_0)\big) = 1\right] \leq e^{\varepsilon} \Pr_{(D_0,D_1)\leftarrow Q}\left[T\big(A(D_1)\big) = 1\right] + \delta + \mathrm{negl}(\lambda)$$

Fingerprinting Codes

Scheme for distributing codewords $w_1, \ldots, w_n$ to n users that can be uniquely traced back to each user, even under collusions of up to k users

Marking assumption asserts that the combined codeword must agree with at least one of the "real" codewords in each position

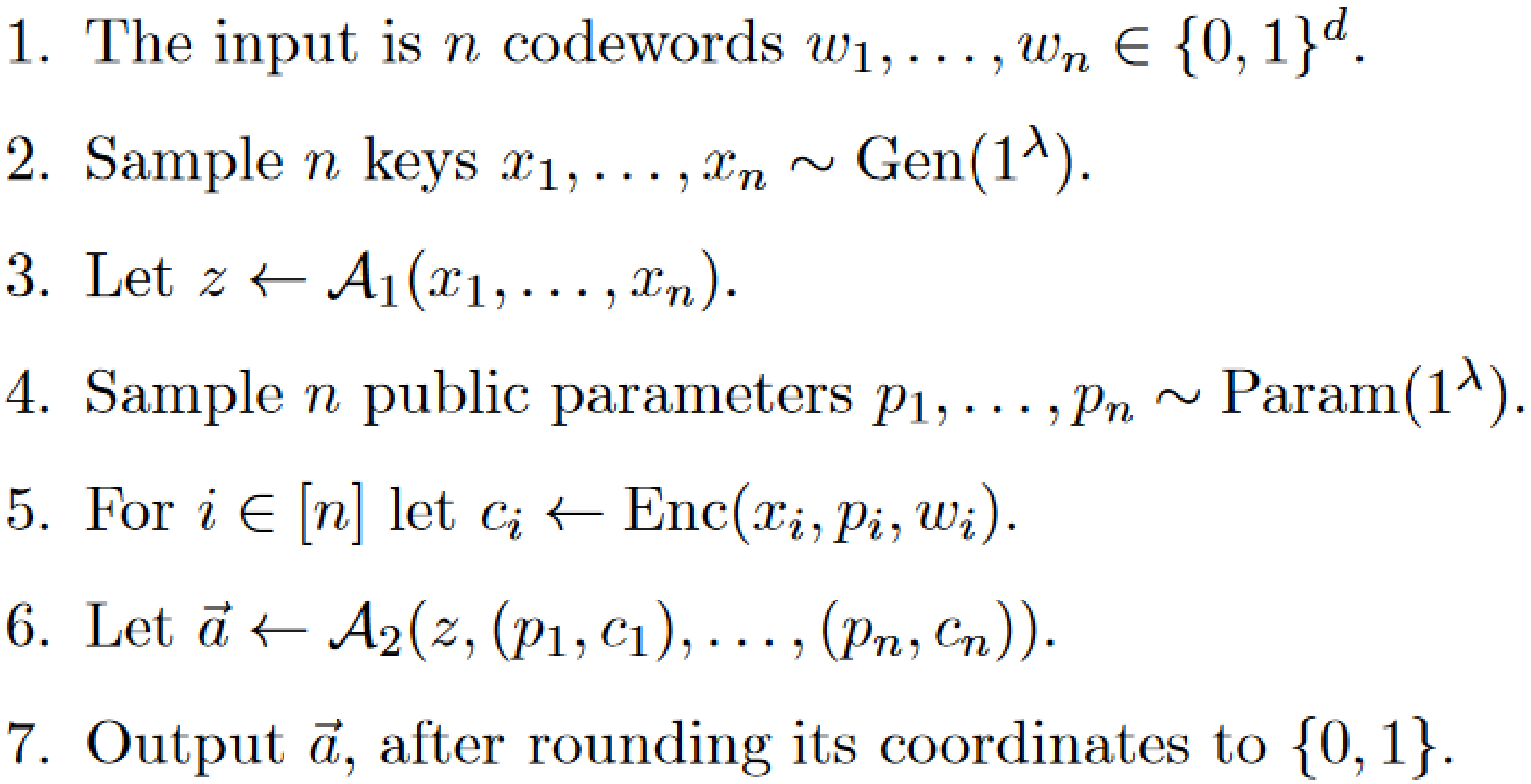
[SU15] For every $k \in [n]$, there is a k-collusion-resilient fingerprinting code of length $d = O(k^2 \log n)$ for n users with failure probability $1 - \frac{1}{n^2}$ and an efficiently computable trace function

CDP Separation

Suppose $A = (A_1, A_2)$ is a $\mathrm{poly}(\Gamma(\lambda))$-time (ε, δ)-CDP algorithm for the DAV problem

Construct an adversary B to fingerprinting code with $\Omega\left(\sqrt{\frac{d}{\log n}}\right)$ colluding users

Adversary to FPC



1. The input is n codewords $w_1, \ldots, w_n \in \{0,1\}^d$.
2. Sample n keys $x_1, \ldots, x_n \sim \mathrm{Gen}(1^\lambda)$.
3. Let $z \leftarrow \mathcal{A}_1(x_1, \ldots, x_n)$.
4. Sample n public parameters $p_1, \ldots, p_n \sim \mathrm{Param}(1^\lambda)$.
5. For $i \in [n]$ let $c_i \leftarrow \mathrm{Enc}(x_i, p_i, w_i)$.
6. Let $\vec{a} \leftarrow \mathcal{A}_2(z, (p_1, c_1), \ldots, (p_n, c_n))$.
7. Output $\vec{a}$, after rounding its coordinates to $\{0,1\}$.

Proof Outline

1. Show B is computationally differentially private w.r.t. the collection of codewords (even though our assumption on A is that it is private w.r.t. the keys)

$$
\begin{aligned}
&\langle r, \mathcal{B}(\vec{w})\rangle \equiv \\
&\equiv \langle r, \mathcal{A}_2\left(\mathcal{A}_1(x_1, \ldots, x_\ell, \ldots, x_n), \vec{p}, \mathrm{Enc}(x_1, p_1, w_1), \ldots, \mathrm{Enc}(x_\ell, p_\ell, w_\ell), \ldots, \mathrm{Enc}(x_n, p_n, w_n)\right)\rangle \\
&\approx_{(\varepsilon,\delta)} \langle r, \mathcal{A}_2\left(\mathcal{A}_1(x_1, \ldots, x_0, \ldots, x_n), \vec{p}, \mathrm{Enc}(x_1, p_1, w_1), \ldots, \mathrm{Enc}(x_\ell, p_\ell, w_\ell), \ldots, \mathrm{Enc}(x_n, p_n, w_n)\right)\rangle \\
&\equiv_c \langle r, \mathcal{A}_2\left(\mathcal{A}_1(x_1, \ldots, x_0, \ldots, x_n), \vec{p}, \mathrm{Enc}(x_1, p_1, w_1), \ldots, \mathrm{Enc}(x_\ell, p_\ell, w'_\ell), \ldots, \mathrm{Enc}(x_n, p_n, w_n)\right)\rangle \\
&\approx_{(\varepsilon,\delta)} \langle r, \mathcal{A}_2\left(\mathcal{A}_1(x_1, \ldots, x_\ell, \ldots, x_n), \vec{p}, \mathrm{Enc}(x_1, p_1, w_1), \ldots, \mathrm{Enc}(x_\ell, p_\ell, w'_\ell), \ldots, \mathrm{Enc}(x_n, p_n, w_n)\right)\rangle \\
&\equiv \langle r, \mathcal{B}(\vec{w'})\rangle.
\end{aligned}
$$

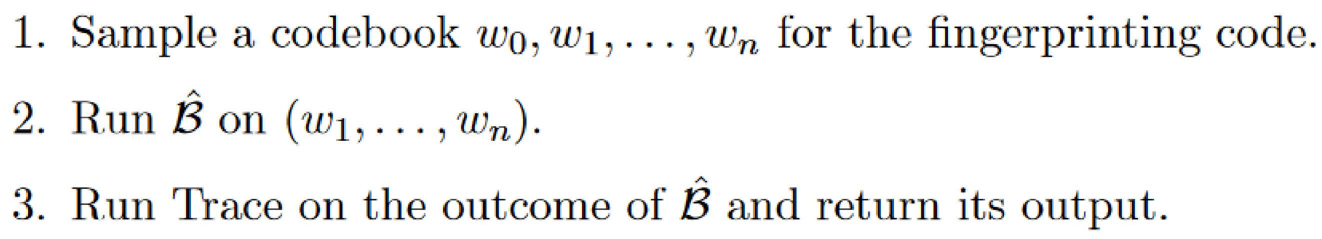
Proof Outline

2. Leveraging the properties of the MILR scheme, show that B effectively ignores most of its inputs, except for at most $\overline{\tau}$ codewords, so B is effectively an FPC adversary that operates on only $\overline{\tau}$ codewords (rather than the n codewords it obtains as input)

1. The input is n codewords $w_1, \ldots, w_n \in \{0,1\}^d$.
2. Sample n keys $x_1, \ldots, x_n \sim \mathrm{Gen}(1^\lambda)$.
3. Let $z \leftarrow \mathcal{A}_1(x_1, \ldots, x_n)$.
4. Sample n public parameters $p_1, \ldots, p_n \sim \mathrm{Param}(1^\lambda)$.
5. Let $J \leftarrow J(\mathcal{A}_1, \vec{x}, z, \vec{p}) \subseteq [n]$ be the subset of coordinates guaranteed to exist by Definition 2.1, of size $|J| = n - \overline{\tau}$.
6. For $i \in J$ let $c_i \leftarrow \mathrm{Enc}(x_i, p_i, 0)$.
7. For $i \in [n] \setminus J$ let $c_i \leftarrow \mathrm{Enc}(x_i, p_i, w_i)$.
8. Let $\vec{a} \leftarrow \mathcal{A}_2(z, (p_1, c_1), \ldots, (p_n, c_n))$.
9. Output $\vec{a}$, after rounding its coordinates to $\{0,1\}$.

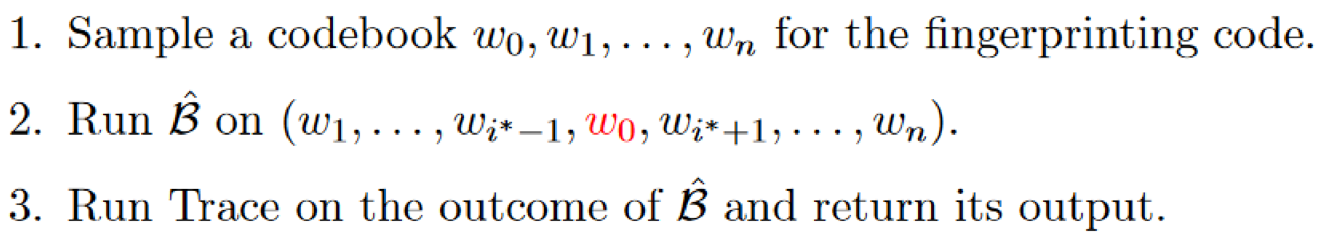
Proof Outline

3. A successful FPC adversary cannot be differentially private, because this would contradict the fact that the tracing algorithm is able to recover one of its input points [BUV14].



1. Sample a codebook $w_0, w_1, \ldots, w_n$ for the fingerprinting code.
2. Run $\hat{\mathcal{B}}$ on $(w_1, \ldots, w_n)$.
3. Run Trace on the outcome of $\hat{\mathcal{B}}$ and return its output.

There exists coordinate exist a coordinate $i^* \neq 0$ that is output with probability at least $\frac{1}{2n}$



1. Sample a codebook $w_0, w_1, \ldots, w_n$ for the fingerprinting code.
2. Run $\hat{\mathcal{B}}$ on $(w_1, \ldots, w_{i^*-1}, w_0, w_{i^*+1}, \ldots, w_n)$.
3. Run Trace on the outcome of $\hat{\mathcal{B}}$ and return its output.

FPC fails with probability at least $\frac{1}{2n}$

Proof Outline

Our gain comes from the fact that B only uses (effectively) τ codewords, and hence, in order to get a contradiction, it suffices to use an FPC with a much shorter codeword-length

Talk Structure

- Multi-instance leakage-resilient (MILR) scheme definition
- Differential privacy separation
- Space bounded adaptive data analysis
- Construction of MILR

Questions?

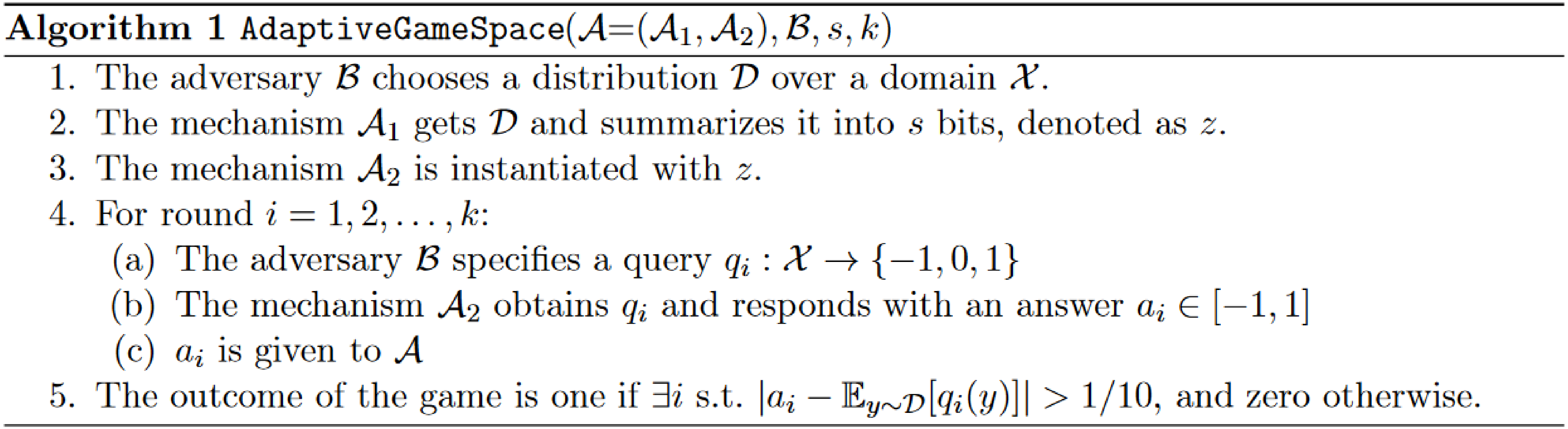
Adaptive Data Analysis

- Given n samples, there exists a computationally efficient oracle that accurately answers $\tilde{O}(n^2)$ adaptive queries [DFH+15]

- There is no computationally efficient oracle that given n samples is accurate on $\tilde{\Omega}(n^2)$ adaptively chosen queries (assuming the existence of one-way functions) [SU15]

Our Results (Adaptive Data Analysis)

Theorem: Every computationally efficient mechanism that is $(0.1, 0.1)$-accurate for k queries must have space complexity at least $\Omega(\sqrt{k})$, assuming the existence of one-way functions



```
Algorithm 1 AdaptiveGameSpace($\mathcal{A}$=($\mathcal{A}_1, \mathcal{A}_2$), $\mathcal{B}$, $s$, $k$)
  1. The adversary $\mathcal{B}$ chooses a distribution $\mathcal{D}$ over a domain $\mathcal{X}$.
  2. The mechanism $\mathcal{A}_1$ gets $\mathcal{D}$ and summarizes it into $s$ bits, denoted as $z$.
  3. The mechanism $\mathcal{A}_2$ is instantiated with $z$.
  4. For round $i = 1, 2, \ldots, k$:
     (a) The adversary $\mathcal{B}$ specifies a query $q_i : \mathcal{X} \to \{-1, 0, 1\}$
     (b) The mechanism $\mathcal{A}_2$ obtains $q_i$ and responds with an answer $a_i \in [-1, 1]$
     (c) $a_i$ is given to $\mathcal{A}$
  5. The outcome of the game is one if $\exists i$ s.t. $|a_i - \mathbb{E}_{y \sim \mathcal{D}}[q_i(y)]| > 1/10$, and zero otherwise.
```

Space Hardness for Adaptive Data Analysis

Theorem: If there exists a $\Gamma(\lambda)$-secure encryption scheme against non-uniform adversaries, then there exists a $\mathrm{poly}(\Gamma(\lambda))$-time adversary B such that:

1. Let $A = (A_1, A_2)$ be a $\mathrm{poly}(\Gamma(\lambda))$-time mechanism with space complexity $s \leq O(\lambda\sqrt{k})$. Then

$$\Pr[AdaptiveGameSpace(A, B, s, k) = 1] > \frac{2}{3}$$

2. Furthermore, the underlying distribution defined by the adversary B can be fully described using $O\left(\lambda\sqrt{k}\right)$ bits, is sampleable in $\mathrm{poly}(\Gamma(\lambda))$- time, and elements sampled from this distribution can be represented using $O(\lambda + \log k)$ bits

Proof Sketch

There exists an adversary B_{sample} that fails every efficient mechanism with sample complexity $t \ll \sqrt{k}$ [SU15]

Use B_{sample} to build an adversary B_{space} that fails every efficient mechanism with space $s \ll \sqrt{k}$

Proof Sketch

B_{sample} uses a uniform distribution over a small set of points of size n hidden to the curator

B_{space} samples n keys $X = (x_1, \ldots, x_n)$ from the MILR scheme and uses a uniform distribution over X, given to A_{space}, who shrinks it into a sketch z of size s bits

For each query q by B_{sample}, define $f_q(x) = q\left(\mathrm{Dec}(x, p_j, c_j)\right)$

Proof Sketch

Would like to claim contradiction, but B_{space} has access to all of X

Define $\hat{B}_{space}$ that only gets to see indices in $[n] \setminus J$, where J has size $n - \overline{\tau}$ and is the set of keys uncompromised by A_{space}

By security of MILR, A_{space} cannot distinguish between $\hat{B}_{space}$ and B_{space}, which leads to a contradiction for $\overline{\tau} \leq t$

MILR Construction

Given an encryption scheme $\Pi' = (\mathrm{Gen}', \mathrm{Enc}', \mathrm{Dec}')$ and $\lambda = \mathrm{poly}(\lambda')$, contrast an MILR scheme as follows:

- Gen: On input 1^λ, return $x \leftarrow_R \{0,1\}^\lambda$
- Param: On input 1^λ, generate a family G of universal hash functions with domain $\{0,1\}^\lambda$ and range $\{0,1\}^{\lambda'}$
- Enc: On input (x, p, m), let $x' = g(x)$ for g described by p and return $\mathrm{Enc}'(x', m)$
- Dec: On input (x, p, c), let $x' = g(x)$ for g described by p and return $\mathrm{Dec}'(x', c)$

k-Bit Fixing Sources

An $(n, 2^\lambda)$-source is a random variable X with range $\left(\{0,1\}^\lambda\right)^n$ and is called k-bit fixing if is fixed on at most k coordinates and uniform on the rest

Closeness to Convex Combination of k-Bit Fixing Sources

Let $F: \left(\{0,1\}^{\lambda}\right)^{n} \rightarrow \{0,1\}^{s}$ be an arbitrary function and $X = (X_1, \dots, X_n) \sim \left(\{0,1\}^{\lambda}\right)^{n}$ and let $Z = F(X)$.

Let H be a family of universal hash functions with domain $\{0,1\}^{\lambda}$ and range $\{0,1\}^{\lambda'}$ and let $G \sim H^{n}$.

There exists a family $V_{G,Z}$ of convex combinations of k-bit fixing $\left(n, 2^{0.1\lambda}\right)$-sources for $k = \frac{2s}{\lambda} + 4$ with

$$\Delta\left[\left(G, Z, G(X)\right), \left(G, Z, V_{G,Z}\right)\right] \leq 2^{-0.1\lambda}$$

Closeness to Convex Combination of k-Bit Fixing Sources

“Even if we give the adversary a leakage $z \in \{0,1\}^s$, hash functions $\vec{g}$ and all the remaining keys, there is a subset of keys that is almost jointly uniformly distributed, i.e., the distribution of the hashed keys $\vec{g}(X)$ is (close to) a convex combinations of k-bit-fixing sources”

Proof uses a variant of the leftover hash lemma

Multi-Security Against Bounded Pre-Processing Adversary

For a fixed k-bit fixing source, the remaining hashed keys are uniformly distributed from the adversary's view, security with respect to these keys follows from the semantic security of the underlying encryption scheme

Applications to Communication Complexity

Suppose ANY sampling based protocol for computing $f(A, B)$ requires k samples $(a_1, b_1), \ldots, (a_k, b_k)$ and $a_i \in \{0,1\}^t$ for each $i \in [k]$

Applications to Communication Complexity

$a_1 \in \{0,1\}^t$

$a_2 \in \{0,1\}^t$

$\vdots$

$a_k \in \{0,1\}^t$

$\longrightarrow$

$b_1 \in \{0,1\}^*$

$b_2 \in \{0,1\}^*$

$\vdots$

$b_k \in \{0,1\}^*$

If a sampling protocol requires $k = \Omega(\eta^2 n)$ samples for success probability $\frac{1}{2} + \frac{\eta}{2}$, then any one-way protocol must use $\Omega(\eta^2 nt)$ communication for success probability $\frac{1}{2} + \eta$

Summary

- Introduce and construct multi-instance leakage resilience scheme
- For the decoded average vector problem, any CDP algorithm requires $\widetilde{\Omega}(\sqrt{d})$ space in the streaming model, while there exists a non-private algorithm that uses $O(\lambda \log d)$ space
- Every computationally efficient mechanism that is $(0.1, 0.1)$-accurate for k queries must have space complexity at least $\Omega(\sqrt{k})$, assuming the existence of one-way functions

Future Directions

Separations for differential privacy and adaptive data analysis without computational assumptions

Separation for differential privacy with a more “natural” problem

Additional applications of MILR

рахмат
danke
謝謝
ngiyabonga
spas
teşekkür ederim
Баярлалаа
mersi
kia ora
barka
welalin
tack
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
спасибі
blagodaram
dank je
misaotra
matondo
paldies
grazzi
thank you
gracias
хвала
nanni
nandri
kiitos
dankie
dhanyavad
maururu
köszönöm
ačiū
asante
manana
bedankt
bayarlalaa
gràçie
hvala
obrigada
murakoze
tenki
enkosi
dankon
akun
djiere dieuf
tau
mochchakkeram
дякую
mamnun
chokrane
dziękuje
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
go raibh maith agat
sobodi
děkuji
sukriya
kop khun krap
sagolun
obrigado
arigatô
takk
dakujem
trugarez
mèsi
didi madloba
najis tuke
ありがとう
tanemirt
grazie
terima kasih
kam sah hamnida
rahmat
rahmet
diolch
dhanyavadagalu
shukriya
mercé
мерси
তোমাকে ধন্যবাদ
감사합니다
xièxie
ευχαριστώ
merci