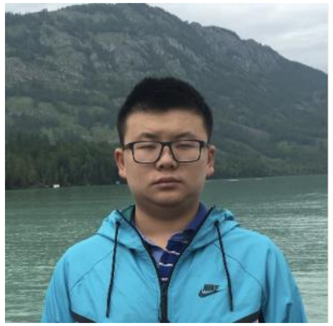
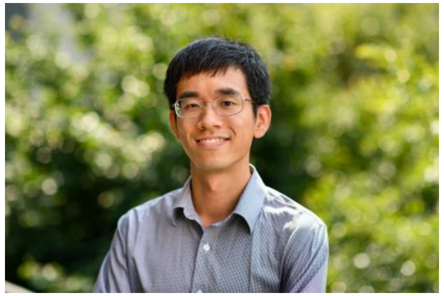
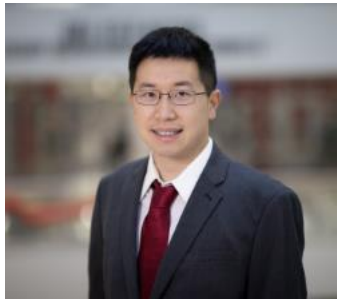
# Lifting Linear Sketches: Optimal Bounds and Adversarial Robustness


Elena Gribelyuk
Princeton

Honghao Lin
CMU

David P. Woodruff
CMU

Huacheng Yu
Princeton

Samson Zhou
Texas A&M

# Streaming Model

## Massive Data Streams

Internet traffics

Sensor networks

Stock markets

# Streaming Model

## Massive Data Streams

Internet traffics

Sensor networks

Stock markets

## Challenges

- Large input space (e.g., $2^{32}$ IPV4 addresses)
- Long input streams (e.g., $10^5$ queries per second)

# Streaming Model

$$a_1, a_2, \cdots, a_m \in [n]$$

- There is an underlying frequency vector $x \in \mathbb{Z}^n$
  - Initialized to $0^n$
  - Updated in each iteration: $x_{a_t} \leftarrow x_{a_t} + 1$, i.e., "*inserting*" $a_t$ into the storage
- Output: Evaluation/approximation of $f(x)$ for a given function $f$
- Goal: Use space *sublinear* in the input space size $n$ and stream length $m$

# Streaming Model

- Examples of function $f$:
  - $\ell_0$ Estimation (Distinct Elements): $f(x) = |\{i : x_i \neq 0\}|$
  - $\ell_p$ Estimation: $f(x) = ||x||_p$
  - $\ell_2$ Heavy Hitters: $f(x) = \{i: |x_i| \geq \varepsilon ||x||_2\}$

- $x$ can also represent other types of input, e.g., matrix or graph

# Example

- Each update of the stream can only increase a coordinate of the frequency vector $x \in \mathbb{R}^n$

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

4 Distinct Elements

# Streaming Model

$$(a_1, w_1), (a_2, w_2), \cdots, (a_m, w_m)$$

- There is an underlying frequency vector $x \in \mathbb{Z}^n$
  - Initialized to $0^n$
  - Updated in each iteration: $x_{a_t} \leftarrow x_{a_t} + w_t$
- Insertion-only stream: when $w_t$ can only be positive
- Insertion-deletion stream : when $w_t$ can be either positive or negative

# Linear Sketch

- Algorithm maintains $Ax$ for a matrix $A$ throughout the stream
  - In the streaming model, the entries of $A$ should be $\mathrm{poly}(n)$ bounded integers and efficiently encoded, e.g., using hash function
- Easy to maintain under additive updates to coordinates of $x$
  - If $\Delta_t$ is the vector of update, we then update the sketch by $A\Delta_t$
- The algorithm then outputs $g(Ax)$ for some post-processing function $g$

# Linear Sketch

A simple example: $(1 \pm \varepsilon)$-approximation of $||x||_2$

- Let $A$ be an $r \times n$ matrix with i.i.d. entries from $\mathrm{Unif}(\{-1, 1\})$
- If $r = O(1/\varepsilon^2)$, with high constant probability,

$$(1 - \varepsilon)||x||_2 \le \frac{1}{\sqrt{r}}||Ax||_2 \le (1 + \varepsilon)||x||_2$$

# Linear Sketch

- Algorithm maintains $Ax$ for a matrix $A$ throughout the stream
  - In the streaming model, the entries of $A$ should be $\mathrm{poly}(n)$ bounded integers

- All insertion-deletion streaming algorithms on a sufficiently long stream might as well be linear sketches [LNW14, AHLW16]

# Linear Sketch

- Lower bounds: for a given task, how many rows do $A$ need to have?

# Linear Sketch

- Lower bounds are fundamental to our understanding of the hardness of streaming problems

- A popular method is to define two "hard" distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ that exhibit a desired gap for the problem of interest

- Then show $d_{TV}(Ax, Ay)$ is small for $x \sim \mathcal{D}_1$ and $y \sim \mathcal{D}_2$ when $A$ has at most $r$ rows

# Linear Sketch

- A simple example: consider the problem of estimating $||x||_2$
- $\mathcal{D}_1 \sim N(0, I_n)$ for a Gaussian distribution with mean zero and identity covariance, and $\mathcal{D}_2 \sim N(0, (1+\varepsilon)I_n)$
- Without loss of generality, assume $A$ has orthonormal rows
- If $x \sim \mathcal{D}_1$, $Ax \sim N(0, I_r)$ while if $y \sim \mathcal{D}_2$, $Ay \sim N(0, (1+\varepsilon)I_r)$
- Using standard results on the number of samples needed to distinguish two normal distributions: $r = \Omega(\log(1/\delta)\ /\varepsilon^2)$

# Linear Sketch

- These techniques imply lower bounds for:
  - $\ell_p$ estimation [GW18]
  - Compressed sensing [PW11, PW13]
  - Eigenvalue estimation and PSD testing [NSW22, PW23]
  - Operator norm and Ky Fan norm [LW16]
  - Norm estimation for adversarially robust streaming algorithms [HW13]
- The distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ are often chosen to be multivariate Gaussians (or somewhat "near" Gaussian), to utilize rotational invariance

# Linear Sketch

- Drawback of these lower bounds: they require the entries of the input vector $x$ to be real-valued as well
  - This is inherent: if $x$ has entries with finite bit complexity, we could use large enough precision entries in $A$ to exactly recover $x$ from $Ax$
- The streaming model is defined on a stream of additive updates to $x$ with finite precision
- These issues mean that none of the above lower bounds actually apply to the data stream model

# Linear Sketch

- This issue has persisted in the literature for several years

- Most of the known discrete lower bounds were obtained via other approaches (e.g., communication complexity)
  - Transfer to discrete linear sketch dimension lower bound by dividing by an $O(\log n)$ factor
  - Can not get optimal bounds in several cases

# Linear Sketch

- Idea: e.g., one could try to discretize the input distribution to the above problem
- Difficulty: the distribution is no longer rotationally invariant, and a priori it is not clear that information about the input is revealed by truncating low order bits
- *Question 1: Is it possible to lift linear sketch lower bounds for continuous inputs to obtain linear sketch lower bounds for discrete inputs?*

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

- Adversarially Robust: "Future queries may depend on previous queries"
- Motivation: Database queries, adversarial ML

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

Attacker

Algorithm

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

$x_1 \leftarrow x_1 + 1$

1

Attacker

Algorithm

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

$$x_1 \leftarrow x_1 + 1$$
$$x_2 \leftarrow x_2 + 1$$

2

Attacker

Algorithm

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

$$x_1 \leftarrow x_1 + 1$$
$$x_2 \leftarrow x_2 + 1$$
$$x_3 \leftarrow x_3 + 1$$

3

Attacker

Algorithm

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

$$x_1 \leftarrow x_1 + 1$$
$$x_2 \leftarrow x_2 + 1$$
$$x_3 \leftarrow x_3 + 1$$
$$x_1 \leftarrow x_1 + 1$$

4

Attacker

Algorithm

# Adversarially Robust Streaming

- Input: Updates to an underlying vector $x$, which arrive sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size $m$ of the input $S$

Attacker

$$x_1 \leftarrow x_1 + 1$$
$$x_2 \leftarrow x_2 + 1$$
$$x_3 \leftarrow x_3 + 1$$
$$x_1 \leftarrow x_1 + 1$$

4


Algorithm

# AMS $F_2$ Algorithm

- Let $s \in \{-1, +1\}^n$ be a random sign vector of length $n$
- Let $Z = \langle s, f \rangle = s_1 f_1 + \cdots + s_n f_n$ and consider $Z^2$

$$E[Z^2] = \sum_{i,j} E\left[s_i s_j f_i f_j\right] = f_1^2 + \cdots + f_n^2$$

$$\mathrm{Var}[Z^2] \leq \sum_{i,j} E\left[s_i s_j s_k s_l f_i f_j f_k f_l\right] \leq 6F_2^2$$

- Take the mean of $O\left(\frac{1}{\varepsilon^2}\right)$ inner products for $(1+\varepsilon)$-approximation [AMS99]

# “Attack” on AMS

- Can learn whether $s_i = s_j$ from $\langle s, e_i + e_j \rangle$
- Let $f_i = 1$ if $s_i = s_1$ and $f_i = -1$ if $s_i \neq s_1$
- $Z = \langle s, f \rangle = s_1 f_1 + \cdots + s_n f_n = m$ and $Z^2 = m^2$ deterministically

- What happened? Randomness of algorithm not independent of input

# Classic Insertion-Only Algorithms

- Space $O\left(\frac{1}{\varepsilon^2} + \log n\right)$ algorithm for $\ell_0$ [KNW10, Blasiok20]
- Space $O\left(\frac{1}{\varepsilon^2} \log n\right)$ algorithm for $\ell_p$ with $p \in (0, 2]$ [BDN17]
- Space $O\left(\frac{1}{\varepsilon^2} n^{1-2/p} \log^2 n\right)$ algorithm for $\ell_p$ with $p > 2$ [Ganguly11,GW18]
- Space $O\left(\frac{1}{\varepsilon^2} \log n\right)$ algorithm for $\ell_2$-heavy hitters [BCINWW17]

# Robust Insertion-Only Algorithms

- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}\log^c n\right)$ algorithm for $\ell_0$ [WZ21]
- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}\log^c n\right)$ algorithm for $\ell_p$ with $p \in (0, 2]$ [WZ21]
- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}n^{1-2/p}\right)$ algorithm for $\ell_p$ with integer $p > 2$ [WZ21]
- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}\log^c n\right)$ algorithm for $\ell_2$-heavy hitters [WZ21]

"No losses* are necessary!"

*up to poly-logarithmic factors

# Robust Insertion-Deletion Streams

For adversarially robust $\ell_p$ estimation:

- By differential privacy, there exists a linear sketch with $r$ rows that is adversarially robust to $\tilde{O}(r^2)$ queries [HKMMSZ20]

- Algorithms with space sublinear in stream length $m$ [BEO22, WZ24]

- For $\mathrm{poly}(n)$ queries, there is no algorithm for constant-factor approximation in sub-linear space in $n$

# Reconstruction Attack on Linear Sketches

- Linear sketches for $\ell_p$ $(p > 0)$ are not robust to adversarial attacks
  - A linear sketch with $r$ rows can be attacked by $\mathrm{poly}(r)$ queries
  - Must use $\Omega(n)$ space to be adversarially robust [HW13]

Algorithm idea [HW13]:

- Iteratively learn sketch matrix $A$
- Then query in the kernel of $A$

# Reconstruction Attack on Linear Sketches

- Attack randomly generates Gaussian vectors
- Analysis uses rotational invariance of Gaussians

Limitations:

- Attack ONLY works on *real-valued inputs*
- ONLY against $\ell_p$ estimation for $p > 0$

# Reconstruction Attack on Linear Sketches

- Recently this was answered for linear sketches for $\ell_0$ in a finite precision stream [GLWYZ24], but techniques specific to $\ell_0$

- *Question 2: Does there exist a sublinear space adversarially robust $\ell_p$-estimation linear sketch in a finite precision stream?*

We give a technique for lifting linear sketch lower bounds for continuous inputs to achieve linear sketch lower bounds for discrete inputs, thus answering the previous open questions

# Upcoming

- Pre-processing for lifting framework

# Questions?

# Discrete Gaussian Distribution

- Let $D(0, S^T S)$ be discrete Gaussian distribution supported on $\mathbb{Z}^n$, with $0^n$ mean and covariance $S^T S$. Then the probability mass function satisfies

$$\Pr_{X \sim D(0, S^T S)}[X = x] \propto \exp\left(-x^T \left(2S^T S\right)^{-1} x\right)$$

- Does not satisfy rotational invariance
- Also has a normalizing constant

# Our Results (Lifting Framework)

Suppose that

- $X \sim D(0, S^T S)$ and $Y \sim N(0, S^T S)$, $Z$ is an arbitrary integer distribution
- $f$ satisfies $\Pr_{x \sim X+Z, y \sim Y+Z}[f(x) \neq f(y)] \leq \frac{\delta}{3}$.
- $g(Ax) = f(x)$ for $x \sim X + Z$ with probability at least $1 - \frac{\delta}{3}$
- $A \in \mathbb{Z}^{r \times n}$ has polynomially-bounded integer entries and the singular value of $S^T S$ is sufficiently large

Then there is another sketching matrix $A' \in \mathbb{R}^{4r \times n}$ with function $h$ such that $h(A'y) = f(y)$ w.p. $1 - \delta$ for $y \sim Y + Z$

# Example Problem ($\ell_2$ Estimation)

- $f(x) = \begin{cases} 0, & \|x\|_2 \leq (1+\varepsilon)N \\ 1, & \|x\|_2 \geq (1+3\varepsilon)N \\ \perp, & \text{otherwise} \end{cases}$

- $X_1 \sim D(0, N^2 I_n)$ and $X_2 \sim D(0, (1+4\varepsilon)^2 N^2 I_n)$
- $Y_1 \sim N(0, N^2 I_n)$ and $Y_2 \sim N(0, (1+4\varepsilon)^2 N^2 I_n)$

- $f$ satisfies $\Pr_{x \sim X_i, y \sim Y_i}[f(x) \neq f(y)] \leq \exp(-cn)$

# Example Problem ($\ell_2$ Estimation)

- Suppose there exists a $g(Ax)$ that can distinguish $X_1$ and $X_2$
- From our theorem, there exists $h(A'x)$ that can distinguish $Y_1$ and $Y_2$
- Then we can use the lower bound for the continuous case!

# Our Results (Applications)

We apply our lifting technique to obtain optimal lower bounds:

| | Existing Real-Valued LB | Previous Discrete LB | Our Discrete LB |
|---|---|---|---|
| $L_p$ Estimation, $p \in [1,2]$ | $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\delta}\right)$ [GW18] | $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\delta}\right)$ [JW13] | $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\delta}\right)$ (Lemma 5.1.2) |
| $L_p$ Estimation, $p > 2$ | $\Omega\left(n^{1-2/p}\log n\right)$ [GW18] | $\Omega\left(n^{1-2/p}\right)$ [LW13, WZ21a] | $\Omega\left(n^{1-2/p}\log n\right)$ (Lemma 5.2.4) |
| Operator Norm | $\Omega\left(\frac{d^2}{\varepsilon^2}\right)$ [LW16] | $\Omega\left(\frac{d}{\log d}\right)$ (folklore) | $\Omega\left(\frac{d^2}{\varepsilon^2}\right)$ (Lemma 5.3.8) |
| Eigenvalue Estimation | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ [NSW22] | $\Omega\left(\frac{1}{\varepsilon^2\log d}\right)$ (folklore) | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ (Theorem 5.4.10) |
| PSD Testing | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ [SW23] | $\Omega\left(\frac{1}{\varepsilon^2\log d}\right)$ (folklore) | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ (Theorem 5.4.11) |
| Compressed Sensing | $\Omega\left(\frac{k}{\varepsilon}\log\frac{n}{k}\right)$ [PW11] | $\Omega\left(\frac{k}{\varepsilon}\right)$ (folklore) | $\Omega\left(\frac{k}{\varepsilon}\log\frac{n}{k}\right)$ (Lemma 5.5.13) |

# Our Results (Adversarial Robustness)

The attack on robust $\ell_p$ estimation is adaptive across multiple rounds, so we cannot apply our theorem directly

Open the procedure of the attack in [HW13], and use the lifting technique in the analysis to obtain an attack using $\mathrm{poly}(r \log n)$ queries to break a discrete sketch

# Our Results (Adversarial Robustness)

- Let $B > 1$ be any fixed desired accuracy parameter

  For any integer sketch with $r$ rows, there exists an algorithm that finds an integer-valued vector on which the sketch fails to output a $B$-approximation to the $\ell_p$ norm of the query

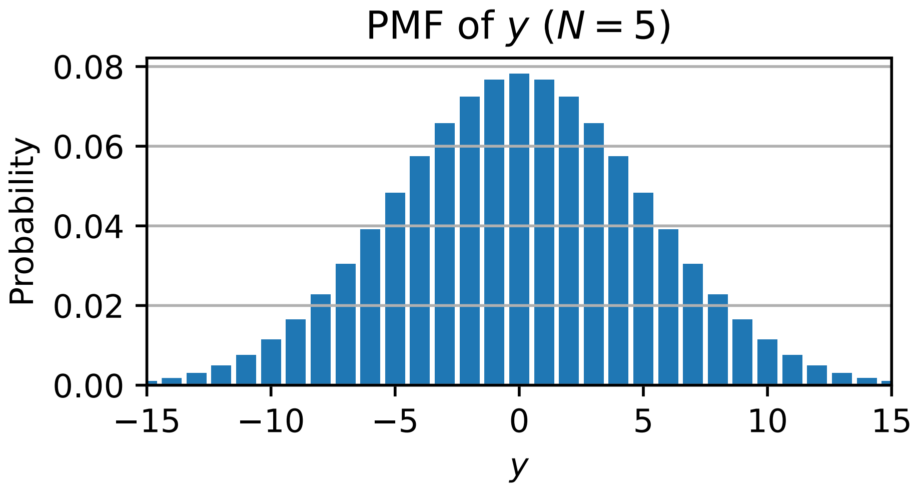
- The adaptive attack uses $\text{poly}(r \log n)$ adaptive queries to the integer sketch and has runtime $\text{poly}(r \log n)$ across $r$ rounds of adaptivity and can be implemented in a polynomially-bounded turnstile stream

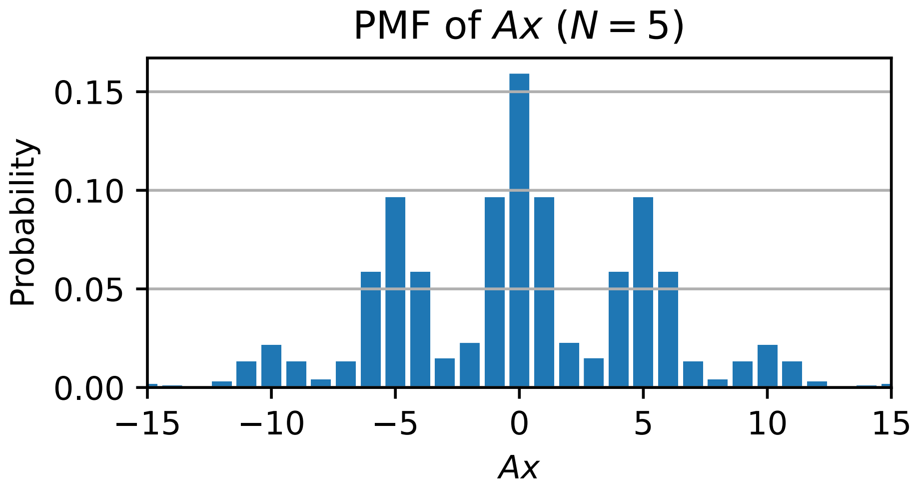
# Technical Overview

- Let $\mathcal{D}_{L,S}$ denote the discrete Gaussian distribution on support $L$ and with covariance matrix $S^T S$
- Suppose $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$ and $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$
- Similar to the continuous case, we want to show the total variation distance between $Ax$ and $y$ is small
- This is not true in general

# Example

Consider $x \sim \mathcal{D}_{\mathbb{Z}^2,\,I_2}$ and $y \sim \mathcal{D}_{A\mathbb{Z}^2,\,A^T}$, where $A = [1 \;\; N]$




Easy to see $y \sim \mathcal{D}_{\mathbb{Z}^2, 1+N^2}$,
(since $AA^T = 1 + N^2$)
"Uniformly" distributed around $O(N)$

$$Ax = x_1 + N x_2$$

Mass concentrates around multiples of $N$.
Low mass at $\frac{N}{2}$

# Lattice Theory Techniques

For $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$ and $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$:

- There exist some bad cases where $Ax$ and $y$ have a large distributional gap
- Under which conditions are the distributions of $Ax$ and $y$ close?
- We address this using lattice theory techniques

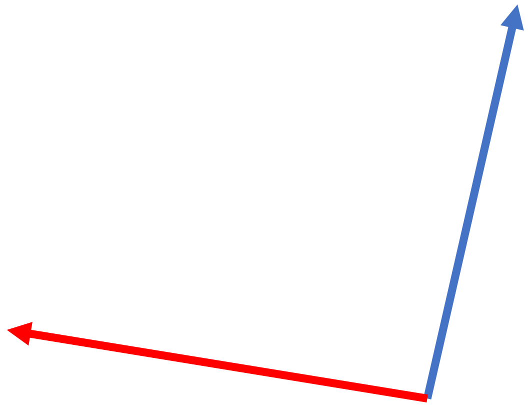
# Lattice Theory Techniques

- The *i*-th successive minima $\lambda_i(\mathcal{L})$ of a lattice $\mathcal{L}$, is defined as the smallest value such that a ball of radius $\lambda_i(\mathcal{L})$ centered at the origin contains at least $i$ linearly independent lattice vectors

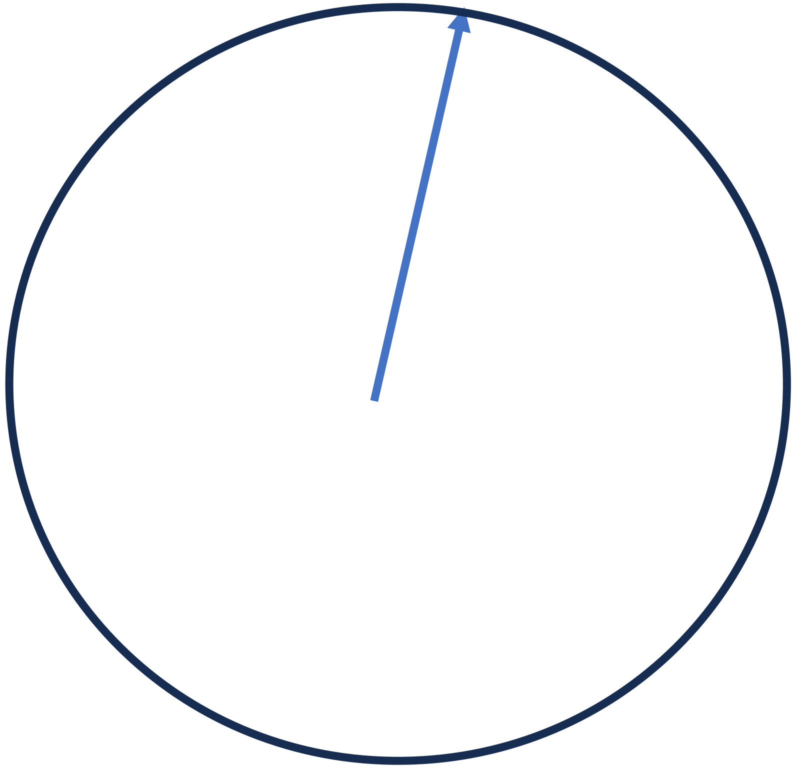
- Let $\mathcal{L}^\perp(A)$ denote the lattice containing integer vectors orthogonal to the rowspan of $A$

# Example

Consider $x \sim \mathcal{D}_{\mathbb{Z}^2, I_2}$ and
$y \sim \mathcal{D}_{A\mathbb{Z}^2, A^T}$, where $A = [1 \;\; N]$



We have $\mathcal{L}^{\perp}(A) = [-N \; 1]$ and
$\lambda_1(\mathcal{L}) = \sqrt{1 + N^2}$



# Lattice Theory Techniques

Thm. (Sufficient condition for small distributional gap [AR16])

Suppose that $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A))\sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$, then

$$1 - 2\varepsilon \leq \frac{\rho_{Ax}(z)}{\rho_y(z)} \leq 1 + 2\varepsilon,$$

where $\rho(z)$ denotes the PMF, $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$, and $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$

# Lattice Theory Techniques

Thm. (Sufficient condition for small distributional gap [AR16])

Suppose that $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A))\sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$, then

$\sigma_n(S)$: The smallest singular value of $S$

where $\rho(z)$ denotes the PMF, $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$, and $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$

# Lattice Theory Techniques

Thm. (Sufficient condition for small distributional gap [AR16])

Suppose that $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A))\sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$, then

$$1 - 2\varepsilon \leq$$

$\mathcal{L}^{\perp}(A)$: The lattice containing integer vectors orthogonal to the rowspan of $A$

where $\rho(z)$ denotes the PMF, $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$, and $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$

# Lattice Theory Techniques

Thm. (Sufficient condition for small distributional gap [AR16])

Suppose that $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A))\sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$, then

where $\rho(z)$ deno

$\lambda_{\max}$: The max successive minima of a lattice

The max successive minima $\lambda_{\max}(\mathcal{L})$ of a lattice $\mathcal{L}$ is the smallest radius such that a ball centered at the origin contains a full basis for the lattice

# Lattice Theory Techniques

Thm. (Sufficient condition for small distributional gap [AR16])

Suppose that $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A))\sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$, then

$$1 - 2\varepsilon \leq \frac{\rho_{Ax}(z)}{\rho_y(z)} \leq 1 + 2\varepsilon,$$

where $\rho(z)$ denotes the PMF, $x \sim \mathcal{D}_{\mathbb{Z}^n, S}$, and $y \sim \mathcal{D}_{A\mathbb{Z}^n, SA^T}$

*If $\varepsilon = \frac{1}{poly(n)}$, the sketch matrix $A$ "passes" through to the covariance*

# Bounding the Successive Minima

- [AR16] requires $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A)) \cdot \sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$
- We can scale $S$ so that $\sigma_n(S) > \mathrm{poly}(n)$
- Hence, we want to upper bound $\lambda_{\max}(\mathcal{L}^{\perp}(A))$ by $\mathrm{poly}(n)$
- However, this is not true in general

# A Simple Example

$$A = \begin{bmatrix} 1 & -2 & 0 & & & \\ 0 & 1 & -2 & \cdots & \cdots & \\ 0 & 0 & 1 & & & \\ & \vdots & & \ddots & \vdots & \\ & & & & -2 & 0 \\ & \cdots & & \cdots & & \\ & & & & 1 & -2 \end{bmatrix}$$

$[2^n, 2^{n-1}, \ldots, 2, 1] \in A^{\perp}$ has exponentially large entries!

# Bounding the Successive Minima

- [AR16] requires $\sigma_n(S) > \lambda_{\max}(\mathcal{L}^{\perp}(A)) \cdot \sqrt{\frac{\ln\left(2n\left(1+\frac{1}{\varepsilon}\right)\right)}{\pi}}$ and we can design $S$ so that $\sigma_n(S) > \mathrm{poly}(n)$

- Key observation: we can add more rows to $A$, as it only makes the sketching matrix stronger

- What to do next: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$

# Bounding the Successive Minima

[Siegel's Lemma]. Let $A \in \mathbb{Z}^{r \times n}$ be a nonzero integer matrix with $r < n$ and entries bounded by $M$. Then there exists a nonzero vector $x$ of integers bounded by $(nM)^{r/(n-r)}$ such that $Ax = 0^r$

- First idea: iteratively add rows to $A$ using Siegel's Lemma.
- What next to do: pre-process $A$ to matrix $A'$ with $r' \geq r$ rows such that $\lambda_{n-r'}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$

# Bounding the Successive Minima

Goal: pre-process $A$ to matrix $A'$ with $r' \geq r$ rows such that $\lambda_{n-r'}(\mathcal{L}^{\perp}(A')) \leq \text{poly}(n)$

- Let $A_0 = A$, and in each time step $t$, find a vector $x_t$ such that $\mathrm{A}_t x_t = 0$ and add $x_t$ to form matrix $A_{t+1}$
- Repeat $0.49n - r$ times. From Siegel's Lemma we have the entries of $x_t$ is bounded by $nM$
- Continue to apply Siegel's Lemma to generate the next $0.51n$ vectors $y_1, y_2, \dots, y_{0.51\mathrm{n}}$ (whose entries are un-bounded)
- Add rows $y_1, y_2, \dots, y_{0.51\mathrm{n}}$ to $A$, form the matrix $A'$

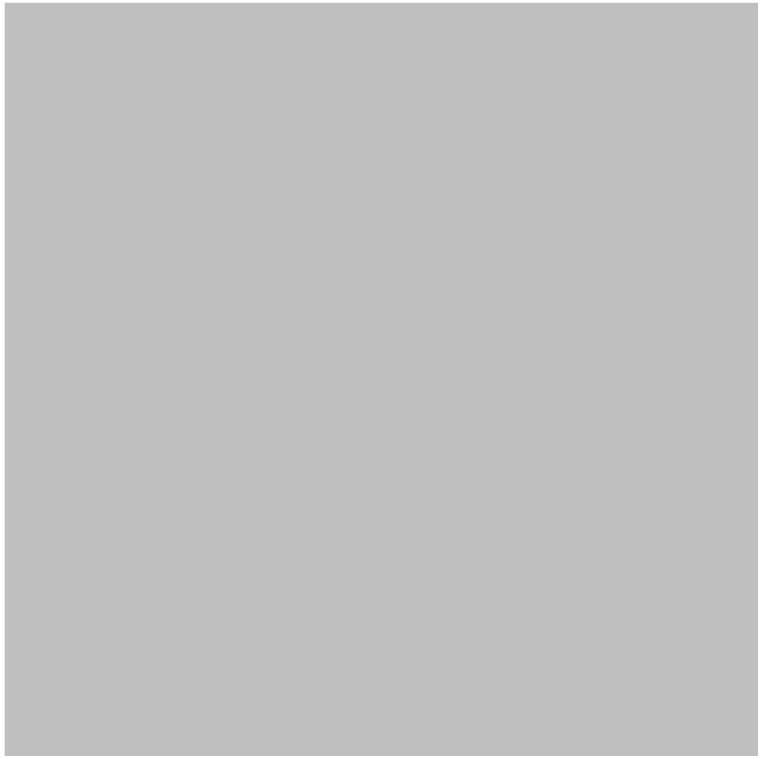
# Bounding the Successive Minima

- First idea: iteratively add rows to $A$ using Siegel's Lemma
- What next to do: pre-process $A$ to matrix $A'$ with $r' \geq r$ rows such that $\lambda_{n-r'}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$

- Then $A'$ satisfies the condition for [AR16] for TVD closeness
- However, $A'$ has $cn + r$ rows, which can not obtain optimal lower bound for some cases
- A better analysis is needed

# Preprocessing

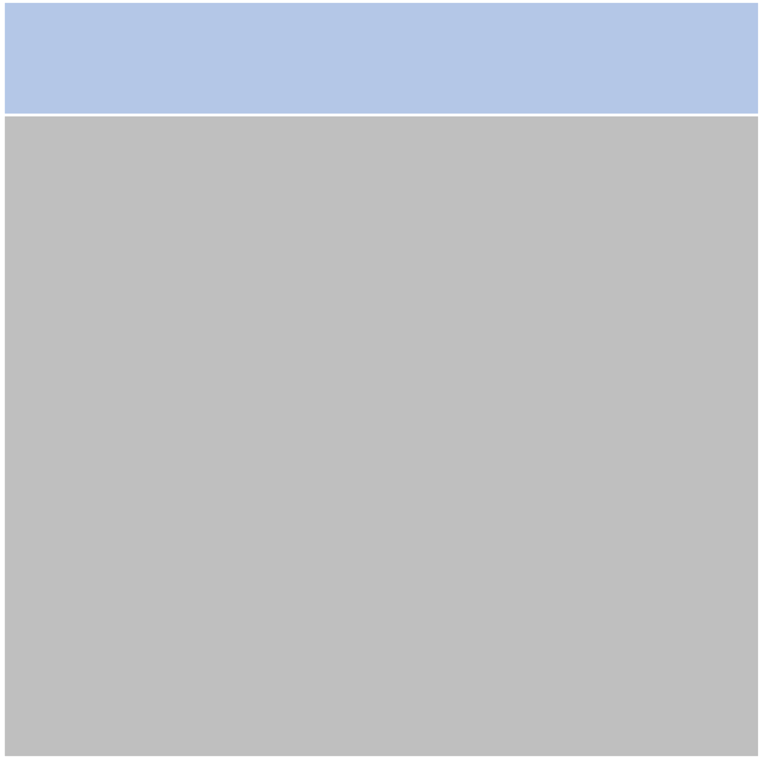
Goal: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$



*Full basis of an $n$-dim space*

# Preprocessing

Goal: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$

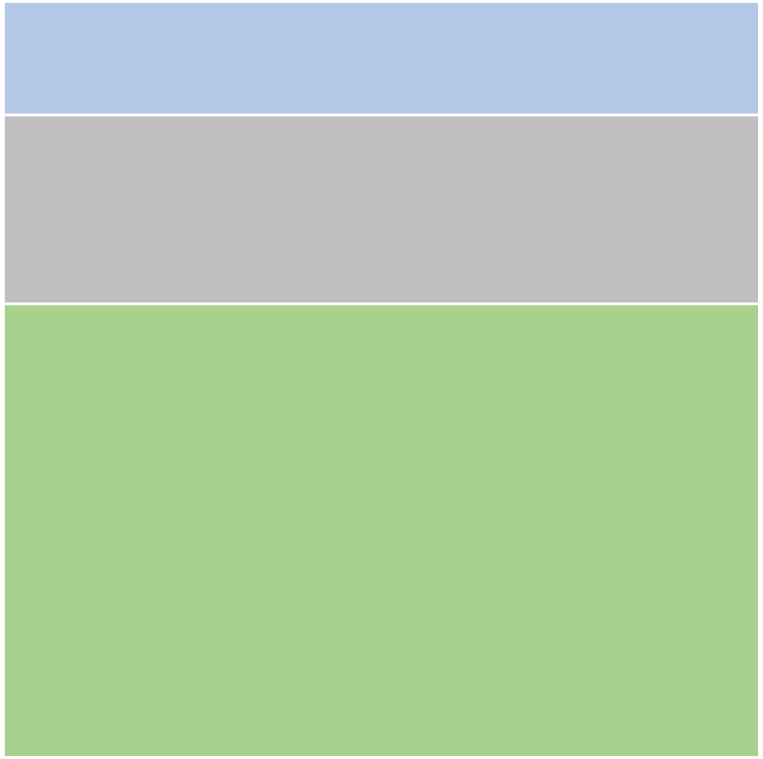


*Row space of* $A$ ($r$ *dimensions*)

*Row space of* $\mathcal{L}^{\perp}(A)$ ($n - r$ *dimensions*)

# Preprocessing

Goal: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$

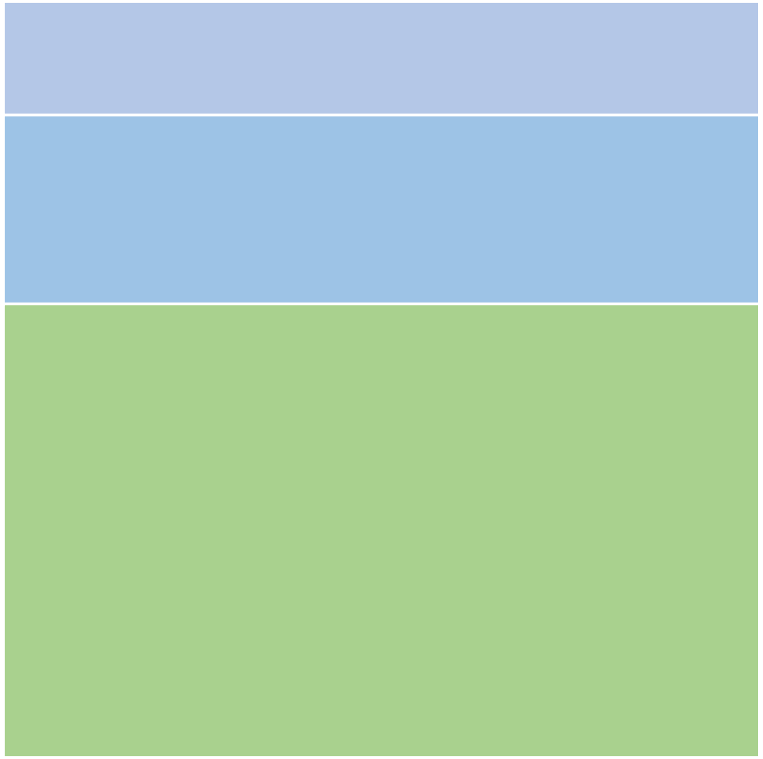


*Row space of* $A$ (*$r$ dimensions*)

***Construct*** *via a probabilistic argument:*
$n - 4r$ *linearly independent integer vectors in* $\mathcal{L}^{\perp}(A)$ *with entries bounded by* $\mathrm{poly}(n)$

# Preprocessing

Goal: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$



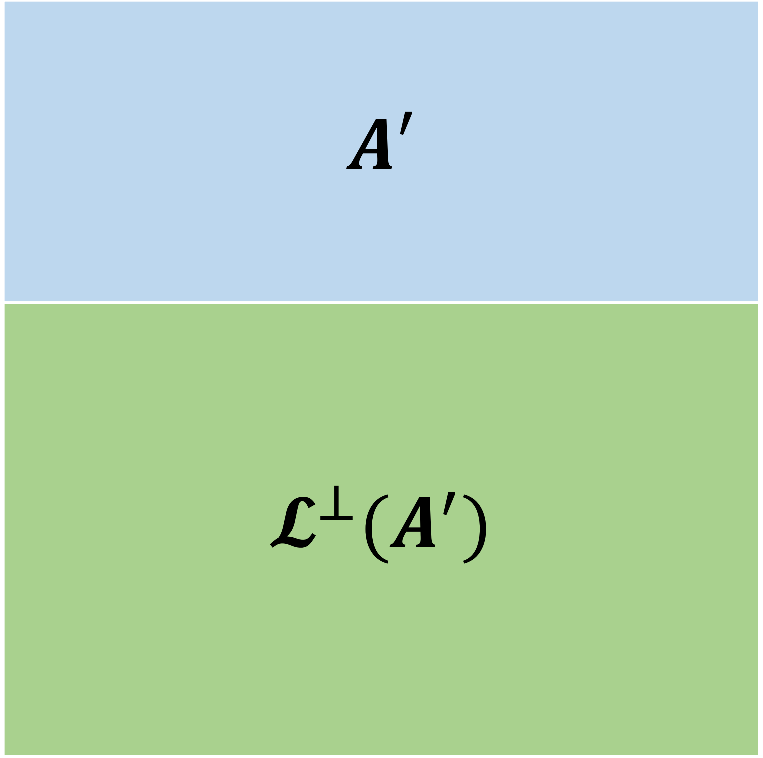
*Row space of $A$ ($r$ dimensions)*

***Construct** iteratively:*
The remaining $3r$ integer vectors

***Construct** via a probabilistic argument:*
*$n - 4r$ linearly independent integer vectors in $\mathcal{L}^{\perp}(A)$ with entries bounded by $\mathrm{poly}(n)$*

# Preprocessing

Goal: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^\perp(A')) \le \mathrm{poly}(n)$

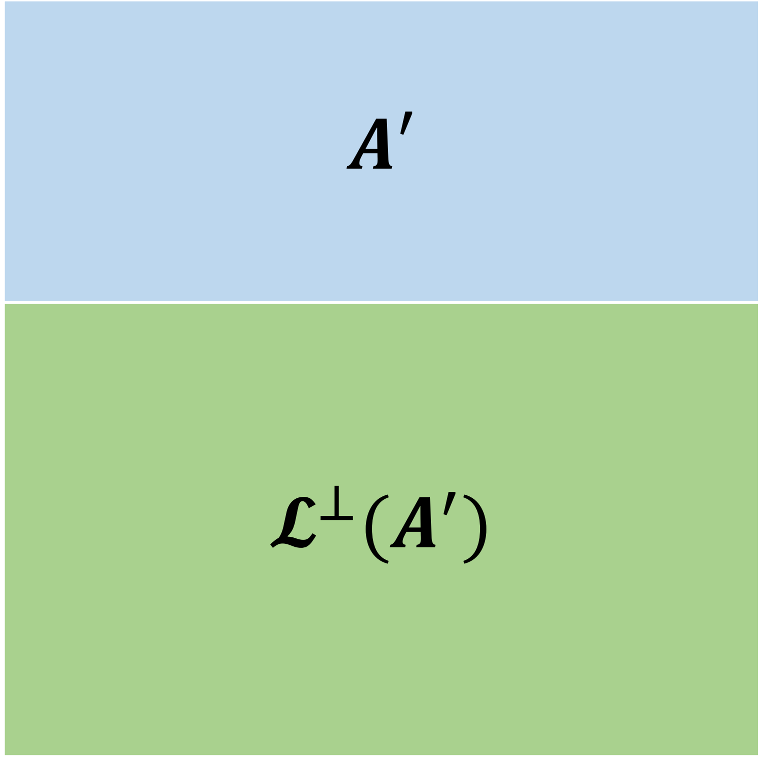



*Row space of $A'$ ($4r$ dimensions)*

***Construct** via a probabilistic argument:*
*$n - 4r$ linearly independent integer vectors in $\mathcal{L}^\perp(A)$ with entries bounded by $\mathrm{poly}(n)$*

# Preprocessing

Goal: pre-process $A$ to matrix $A'$ with $r' = O(r)$ rows such that $\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$




*Row space of $A'$ ($4r$ dimensions)*

***Construct via a probabilistic argument:***
*$n - 4r$ linearly independent integer vectors in $\mathcal{L}^{\perp}(A')$ with entries bounded by $\mathrm{poly}(n)$*

By definition of successive minima:
$\lambda_{\max}(\mathcal{L}^{\perp}(A')) \leq \mathrm{poly}(n)$

# Preprocessing

- We show that we can generate $n - O(r)$ linearly independent integer vectors in $\mathcal{L}^{\perp}(A)$ with entries bounded by $\mathrm{poly}(n)$
- Probabilistic argument: Suppose we have found $t$ such vectors
- Let $B \in \mathbb{R}^{(n-t) \times n}$ denote the matrix whose rows form a basis of the orthogonal complement to the span of these $t$ vectors

# Probabilistic Argument for Preprocessing

- Randomly pick $s = n^{O(r)}$ vectors $v^i$ with entries in $\{0, 1, 2, \ldots, M-1\}$, for sufficiently large $M = \mathrm{poly}(n)$
- Event (i): There exists $1 \leq i < j \leq s$ such that $Av^i = Av^j$
- Event (i) holds with high probability: the entries of $Av^i$ are bounded by $\mathrm{poly}(n)$, so use birthday paradox

# Probabilistic Argument for Preprocessing

- Randomly pick $s = n^{O(r)}$ vectors $v^i$ with entries in $\{0, 1, 2, \dots, M-1\}$, for sufficiently large $M = \text{poly}(n)$

- Event (ii): For all $1 \leq i < j \leq s$ we have $Bv^i \neq Bv^j$
- Event (ii) holds with high probability: W.L.O.G., we can write $B$ in reduced row echelon form
- Then for every $v^i, v^j$ we have $\Pr\left[B\left(v^i - v^j\right) = 0\right] \leq \left(\frac{1}{M}\right)^{n-t}$
- Take a union bound over all $i, j$

# Bounding the Successive Minima

- Randomly pick $s = n^{O(r)}$ vectors $v^i$ with entries in $\{0, 1, 2, \ldots, M-1\}$, for sufficiently large $M = \mathrm{poly}(n)$

- Conditioning on events (i) and (ii) holding, then $v^i - v^j$ is the vector we need
  - It is in kernel of $A$ so in orthogonal lattice
  - It is not in kernel of rows in orthogonal lattice already found

- We can iteratively apply this argument until $t = n - O(r)$

# Bounding the Successive Minima

- Suppose we have chosen these $n - O(r)$ vectors
- Iteratively generate $O(r)$ integer vectors that are orthogonal to both the row span $A$ of and the $n - 4r$ integer vectors
- Add these $O(r)$ vectors to rows of $A$ and form a new matrix $A'$ with $O(r)$ rows, which satisfies the requirement

# Upcoming

- Cell lemma and lifting framework

# Questions?

# Cell Lemma

Recall $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$, $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$, and $z \sim N(0, S^T S)$

Can assume $\mathcal{L}^{\perp}(A)$ has bounded successive minima

Let $\eta$ be a uniform noise in one unit cell of the lattice $A\mathbb{Z}^n$

Goal:

Distribution of $Ax + \eta$ *Close to* Distribution of $Az$

*≈ Discrete sketch*

*Continuous sketch*

# Cell Lemma

Recall $x \sim \mathcal{D}_{\mathbb{Z}^n,S}$, $y \sim \mathcal{D}_{A\mathbb{Z}^n,SA^T}$, and $z \sim N(0, S^T S)$

Can assume $\mathcal{L}^{\perp}(A)$ has bounded successive minima

Let $\eta$ be a uniform noise in one unit cell of the lattice $A\mathbb{Z}^n$

Distribution of $Ax + \eta$

[AR16] $\approx_{1+1/\mathrm{poly}(n)}$

Distribution of $y + \eta$

$\approx_{1+1/\mathrm{poly}(n)}$

$N(0, A^T S^T S A)$.

=

Distribution of $Az$

# Cell Lemma

Distribution of $Ax + \eta$ $\xrightarrow{\textit{Close to}}$ Distribution of $Az$

$\approx$ *Discrete sketch* | *Continuous sketch*

Real discrete sketches take $Ax$ as inputs. Why is the above useful?

# Cell Lemma

Distribution of $Ax + \eta$ — *Close to* → Distribution of $Az$

$\approx$ *Discrete sketch* — *Continuous sketch*

Real discrete sketches take $Ax$ as inputs. Why is the above useful?

- $Ax$ can be recovered from $Ax + \eta$ by rounding
- The rounding operation can be baked into post-processing:

$$g(Ax) = g \circ \mathrm{round}(Ax + \eta)$$

# Cell Lemma

Distribution of $Ax + \eta$ — *Close to* → Distribution of $Az$

$\approx$ *Discrete sketch*

*Continuous sketch*

Real discrete sketches take $Ax$ as inputs. Why is the above useful?

- $Ax$ can be recovered from $Ax + \eta$ by rounding
- The rounding operation can be baked into post-processing:

$$g(Ax) = g \circ \mathrm{round}(Ax + \eta)$$

- Thus, we can assume the (discrete) algorithm takes $Ax + \eta$ as inputs
- So it should also work for $Az$ (continuous input)

# Our Results (Applications)

We apply our lifting technique to obtain optimal lower bounds:

| | Existing Real-Valued LB | Previous Discrete LB | Our Discrete LB |
|---|---|---|---|
| $L_p$ Estimation, $p \in [1,2]$ | $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\delta}\right)$ [GW18] | $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\delta}\right)$ [JW13] | $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\delta}\right)$ (Lemma 5.1.2) |
| $L_p$ Estimation, $p > 2$ | $\Omega\left(n^{1-2/p}\log n\right)$ [GW18] | $\Omega\left(n^{1-2/p}\right)$ [LW13, WZ21a] | $\Omega\left(n^{1-2/p}\log n\right)$ (Lemma 5.2.4) |
| Operator Norm | $\Omega\left(\frac{d^2}{\varepsilon^2}\right)$ [LW16] | $\Omega\left(\frac{d}{\log d}\right)$ (folklore) | $\Omega\left(\frac{d^2}{\varepsilon^2}\right)$ (Lemma 5.3.8) |
| Eigenvalue Estimation | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ [NSW22] | $\Omega\left(\frac{1}{\varepsilon^2\log d}\right)$ (folklore) | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ (Theorem 5.4.10) |
| PSD Testing | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ [SW23] | $\Omega\left(\frac{1}{\varepsilon^2\log d}\right)$ (folklore) | $\Omega\left(\frac{1}{\varepsilon^4}\right)$ (Theorem 5.4.11) |
| Compressed Sensing | $\Omega\left(\frac{k}{\varepsilon}\log\frac{n}{k}\right)$ [PW11] | $\Omega\left(\frac{k}{\varepsilon}\right)$ (folklore) | $\Omega\left(\frac{k}{\varepsilon}\log\frac{n}{k}\right)$ (Lemma 5.5.13) |

# Application: $\ell_p$ Norm Estimation, $p \in [1,2]$

- $f(x) = \begin{cases} 0, & \|x\|_2 \leq (1+\varepsilon)N \\ 1, & \|x\|_2 \geq (1+3\varepsilon)N \\ \perp, & \text{otherwise} \end{cases}$

- $Y_1 \sim N(0, N^2 I_n)$ vs. $Y_2 \sim N(0, (1+4\varepsilon)^2 N^2 I_n)$

- With high probability, $\Pr_{x \sim Y_1}[f(x) = 1]$ and $\Pr_{y \sim Y_2}[f(y) = 0]$

# Application: $\ell_p$ Norm Estimation, $p \in [1,2]$

- With high probability, $\Pr_{x \sim Y_1}[f(x) = 1]$ and $\Pr_{y \sim Y_2}[f(y) = 0]$

- However, $d_{TV}(Ax, Ay) \leq 1 - \delta$ unless $A$ has sketching dimension $\Omega\left(\frac{1}{\varepsilon^2} \log \frac{1}{\delta}\right)$

- Our technique recovers the same bound for integer sketches

# Application: $\ell_p$ Norm Estimation, $p > 2$

- $\mathcal{D}_1 = N(0, N^2 I_n)$
- $\mathcal{D}_2 = N(0, N^2 I_n) + \sum_{i \in [T]} \Theta\left(\frac{\varepsilon^{1/p} N n^{1/p}}{t^{1/p}}\right) e_i$, where $T$ is a random set of $O\left(\log \frac{1}{\delta}\right)$ coordinates of $[n]$

- With high probability, $\Pr_{x \sim \mathcal{D}_1}\left[\|x\|_p \leq (1 + 2\varepsilon) N\beta\right]$ and $\Pr_{y \sim \mathcal{D}_2}\left[\|y\|_p \geq (1 + 4\varepsilon) N\beta\right]$

# Application: $\ell_p$ Norm Estimation, $p > 2$

- With high probability, $\Pr_{x \sim \mathcal{D}_1}\left[\|x\|_p \leq (1+2\varepsilon)N\beta\right]$ and $\Pr_{y \sim \mathcal{D}_2}\left[\|y\|_p \geq (1+4\varepsilon)N\beta\right]$

- However, $d_{TV}(Ax, Ay) \leq 1 - \delta$ unless $A$ has sketching dimension $\Omega\left(n^{1-2/p} \frac{1}{\varepsilon^{-2/p}} \log n \log^{2/p} \frac{1}{\delta}\right)$

- Our technique recovers the same bound for integer sketches

# Reconstruction Attack on Linear Sketches

- Linear sketches for $\ell_p$ estimation ($p > 0$) are "not robust" to adversarial attacks, require $\Omega(n)$ dimension [HW13]
- Approximately learn sketch matrix $A$, query something in the kernel of $A$
- Iterative process, start with $V_1 = \emptyset, \ldots, V_r$
- Correlation finding: Find vectors weakly correlated with $A$ orthogonal to $V_{i-1}$
- Boosting: Use these vectors to find strongly correlated vector $v$
- Progress: Set $V_i = \mathrm{span}(V_{i-1}, v)$

# Correlation Finding

- Start with a subspace $V_i$, iterate over small increments of $\sigma^2$

- Sample $v_1, \ldots, v_m \sim \mathcal{D}_{\mathbb{Z}^n, \Sigma_{\sigma^2}}$, where $\Sigma_{\sigma^2}$ is a covariance matrix that projects onto $V_i^{\perp}$ and scaled by $\sigma^2$, up to some small noise ($m = \text{poly}(n)$)

- Let $v'_1, \ldots, v'_{m'}$ be the positively labeled samples, i.e., $\mathbb{A}(v'_i) = 1$

# Boosting and Progress

- Let $v_\sigma = \text{argmax}_u \sum \langle u, v_i' \rangle^2$
- If $\frac{1}{m'} \cdot \sum \langle v_\sigma, v_i' \rangle^2 \geq \sigma^2 + \Delta$ for some gap $\Delta$, add $v^*$ to $V_i$, where $v^*$ is the part of $v_\sigma$ orthogonal to $V_i$

**Input:** Oracle $\mathcal{A}$ providing access to a function $f : \mathbb{R}^n \to \{0,1\}$, parameters $B \geq 4$, and sufficiently large $\alpha = \text{poly}(n)$ satisfying $\alpha \geq \ell_{\mathbf{A}}^2 \cdot \frac{\ln(2n(1+1/\varepsilon))}{\pi}$ after pre-processing via Lemma 3.1, for all possible integer matrices $\mathbf{A} \in \mathbb{Z}^{r \times n}$ initially with $\text{poly}(n)$-bounded entries.

**Attack:** Let $V_0 = \emptyset$, $m = \mathcal{O}\left(B^{13} n^{11} \log^{15}(n)\right)$, $S = [\alpha, \alpha \cdot B] \cap \zeta\mathbb{Z}$ where $\zeta = \frac{1}{20(Bn)^2 \log(Bn)}$.

**For** $t \in [r+1]$**:**

(1) For each $\sigma^2 \in S$:

(a) Sample $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim D(V^\perp, \sigma^2)$. Query $\mathcal{A}$ on each $\mathbf{x}_i$ and let $a_i = \mathcal{A}(\mathbf{x}_i)$.

(b) Let $s(t, \sigma^2) = \frac{1}{m}\sum_{i=1}^m a_i$ denote the fraction of samples that are positively labeled.

i. If either (1) $\sigma^2 \geq \alpha \cdot B/2$ and $s(t,\sigma^2) \leq 1 - \zeta$ or (2) $\sigma^2 \leq 2 \cdot \alpha$ and $s(t,\sigma^2) \geq \zeta$, then terminate and **return** $(V_t^\perp, \sigma^2)$.

ii. Else let $\mathbf{x}'_1, \ldots, \mathbf{x}'_{m'}$ be the vectors such that $\mathcal{A}(\mathbf{x}'_i) = 1$ for all $i \in [m']$.

(c) If $m' < \frac{m}{100B^2 n}$, increment $\sigma^2$. Else, compute $v_\sigma = \text{argmax}_{\mathbf{v} \in \mathbb{R}^n} z(\mathbf{v})$ for $z(\mathbf{v}) = \frac{1}{m'}\sum_{i=1}^{m'} \langle \mathbf{v}, \mathbf{x}'_i \rangle^2$

(2) Let $\mathbf{v}'$ represent the first vector $\mathbf{v}_\sigma$ with $z(\mathbf{v}_\sigma) \geq \sigma^2 + \frac{\sigma^2}{4} + \frac{1}{14Br}$.

(a) If no such $\mathbf{v}_\sigma$ was found, set $V_{t+1} = V_t$ and proceed to the next round.

(b) Otherwise, let $\mathbf{v}^* = \mathbf{v}'$. Compute $\mathbf{v}_t = \mathbf{v}^* - \frac{\sum_{\mathbf{v} \in V_t} \mathbf{v}\langle \mathbf{v}, \mathbf{v}^* \rangle}{\left\| \sum_{\mathbf{v} \in V_t} \mathbf{v}\langle \mathbf{v}, \mathbf{v}^* \rangle \right\|_2}$ and set $V_{t+1} = V_t \cup \{\mathbf{v}_t\}$.

Fig. 1: Algorithm that creates an adaptive attack using a turnstile data stream.

# Conditional Expectation Lemma

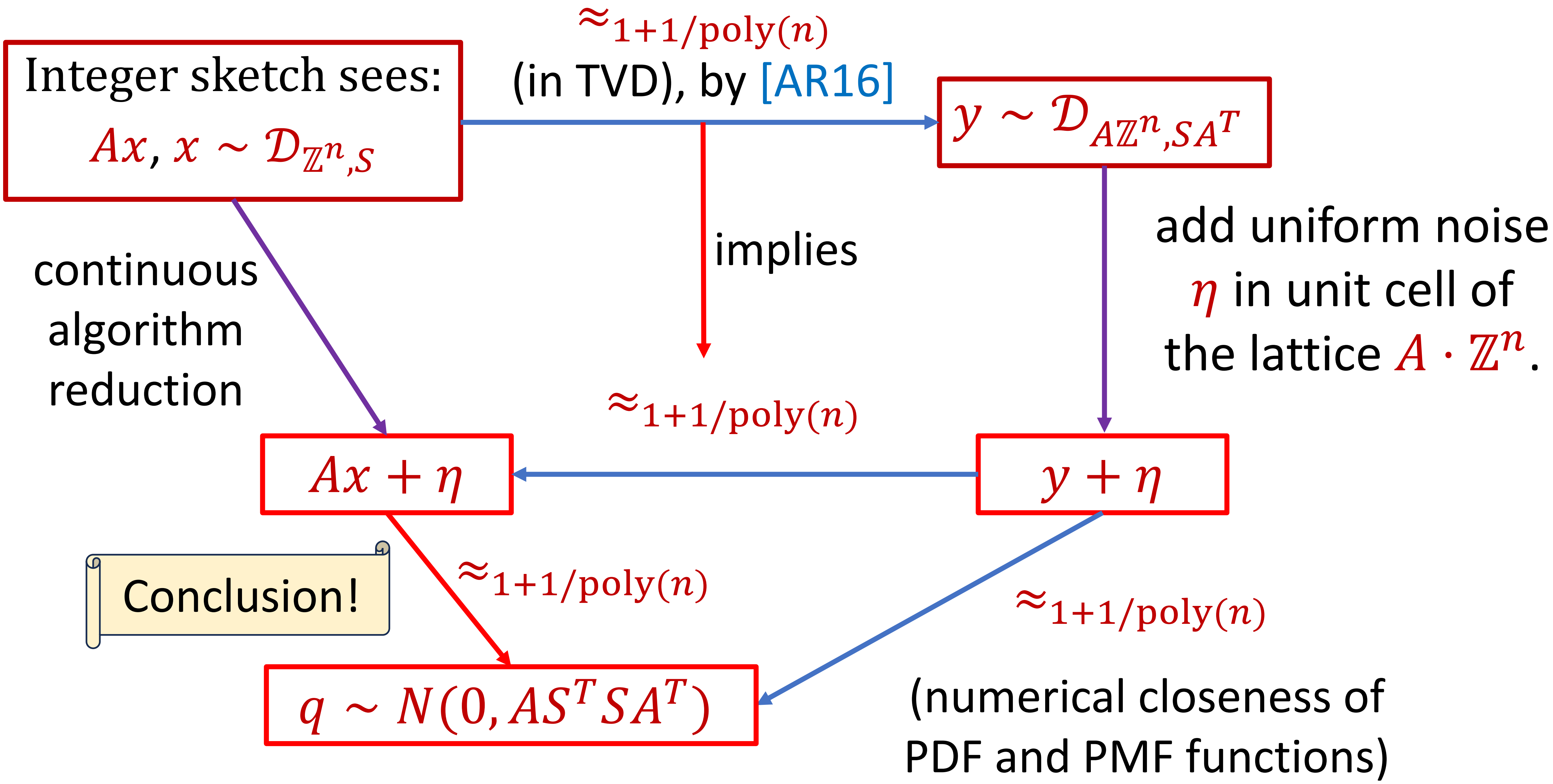
- There exists a variance $\sigma^2$ and a vector $u \in A \cap V_i^\perp$ such that for $x \sim \mathcal{D}_{\mathbb{Z}^n,\Sigma_{\sigma^2}}$,

$$\mathbb{E}[\langle u, x\rangle^2 | \mathbb{A}(x) = 1] \geq \mathbb{E}[\langle u, x\rangle^2] + \Delta$$

- Proof by lifting to continuous version of conditional expectation lemma [HW13]

# Summary

- We give a framework for “lifting” lower bound techniques for linear sketches to integer sketches
- Idea is to use discrete Gaussians in place of a continuous Gaussian on “well-behaved” sketches
- Can be used to achieve lower bounds for a range of problems, including adversarial robust norm estimation


Integer sketch sees:
$Ax, x \sim \mathcal{D}_{\mathbb{Z}^n, S}$
$\approx_{1+1/\mathrm{poly}(n)}$
(in TVD), by [AR16]
$y \sim \mathcal{D}_{A\mathbb{Z}^n, SA^T}$
implies
continuous
algorithm
reduction
add uniform noise
$\eta$ in unit cell of
the lattice $A \cdot \mathbb{Z}^n$.
$\approx_{1+1/\mathrm{poly}(n)}$
$Ax + \eta$
$y + \eta$
Conclusion!
$\approx_{1+1/\mathrm{poly}(n)}$
$\approx_{1+1/\mathrm{poly}(n)}$
$q \sim N(0, AS^T SA^T)$
(numerical closeness of
PDF and PMF functions)

Cell Lemma

# Future Directions


- Lower bounds for streaming beyond integer linear sketches?

Attacks on linear-sketches for $\ell_0$ estimation [GLWYZ24]

Attacks on streaming algorithms for $\ell_0$ estimation

Attacks on linear-sketches for $\ell_p$ estimation [This talk]

Attacks on streaming algorithms for $\ell_p$ estimation