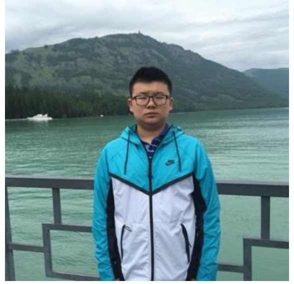
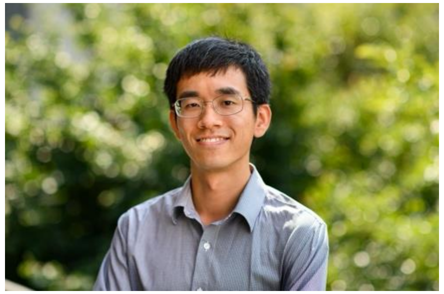
Lifting Linear Sketches: Optimal Bounds and Adversarial Robustness

Elena Gribelyuk
Princeton University



Honghao Lin
CMU & Google

David P. Woodruff
CMU



Huacheng Yu
Princeton University

Samson Zhou
Texas A&M

Streaming Model

- Input: We assume there is an underlying frequency vector $x \in \mathbb{Z}^n$, initialized to 0^n
- Update: The stream consists of updates of the form (i_t, w_t), meaning $x_{i_t} \leftarrow x_{i_t} + w_t$
- Output: Evaluation (or approximation) of $f(x)$ for a given function f
- Goal: Use space *sublinear* in the size n and m of the stream length

Streaming Model

- Insertion-Only model: when w_t can only be positive

- Turnstile model: when w_t can be both 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 *poly(n)* bounded integers
- Easy to maintain under additive updates to coordinates of *x*
- The algorithm then outputs $f(Ax)$ for some post-processing function f
- All turnstile streaming algorithms on a sufficiently long stream might as well be linear sketches [LNW14, AHLW16]

Linear Sketch

- Lower bounds are fundamental to our understanding of the capabilities and limitations of streaming algorithms

- A popular method is to define two "hard" distribution $\mathcal{D}_1$ 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 $\mathrm{y} \sim \mathcal{D}_2$ 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 $x \sim \mathcal{D}_1$, $\mathrm{A}x \sim N(0, I_r)$ while if $\mathrm{y} \sim \mathcal{D}_2$, $\mathrm{A}y \sim N(0, (1+\varepsilon)I_r)$
- 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:
  - ℓ_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

- 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: 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 dimension *n* of the input x

$x_1 \leftarrow x_1 + 1$

1

Estimate number of non-zero coordinates of *x*

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 dimension *n* of the input x

$$x_1 \leftarrow x_1 + 1$$
$$x_4 \leftarrow x_4 + 1$$

2

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 dimension *n* of the input x

$x_1 \leftarrow x_1 + 1$

$x_4 \leftarrow x_4 + 1$

$x_2 \leftarrow x_2 + 1$

3

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 dimension *n* of the input x

$$x_1 \leftarrow x_1 + 1$$
$$x_4 \leftarrow x_4 + 1$$
$$x_2 \leftarrow x_2 + 1$$
$$x_1 \leftarrow x_1 + 1$$

4

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 dimension *n* of the input x

$x_1 \leftarrow x_1 + 1$

$x_4 \leftarrow x_4 + 1$

$x_2 \leftarrow x_2 + 1$

$x_1 \leftarrow x_1 + 1$

4

Classic Insertion-Only Algorithms

- Space $O\left(\frac{1}{\varepsilon^2} + \log n\right)$ algorithm for ℓ_0 [KNW10, Blasiok20]
- Space $O\left(\frac{1}{\varepsilon^2} \log n\right)$ algorithm for ℓ_p with $p \in (0, 2]$ [BDN17]
- Space $O\left(\frac{1}{\varepsilon^2} n^{1-2/p} \log^2 n\right)$ algorithm for ℓ_p with $p > 2$ [Ganguly11,GW18]
- Space $O\left(\frac{1}{\varepsilon^2} \log n\right)$ algorithm for ℓ_2-heavy hitters [BCINWW17]

Robust Insertion-Only Algorithms

- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}\log n\right)$ algorithm for ℓ_0
- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}\log n\right)$ algorithm for ℓ_p with $p \in (0, 2]$
- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}n^{1-2/p}\right)$ algorithm for ℓ_p with integer $p > 2$
- Space $\tilde{O}\left(\frac{1}{\varepsilon^2}\log n\right)$ algorithm for L_2-heavy hitters

"No losses* are necessary!"

- However, large gap between upper and lower bounds for turnstile streams: $\tilde{O}(n)$ upper bound, $\Omega(polylog(n))$ lower bound.

Reconstruction Attack on Linear Sketches

- Linear sketches for ℓ_p estimation ($p > 0$) are "not robust" to adversarial attacks, require $\Omega(n)$ dimension [Hardt-Woodruff13]
- Approximately learn sketch matrix A, then query $x \in Ker(A)$ or $x = 0^n$ each with probability ½
- 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 $V_i = \mathrm{span}(V_{i-1}, v)$

Reconstruction Attack on Linear Sketches

- Attack randomly generates Gaussian vectors
- Analysis uses rotational invariance of Gaussians
- Attack ONLY works on *real-valued inputs*
- *Question: Does there exist a sublinear space adversarially robust F_2-estimation linear sketch in a finite precision stream?*
- Recently this was answered for linear sketches for ℓ_0 in a finite precision stream [Gribelyuk-Lin-Woodruff-Yu-Zhou24]. Techniques specific to ℓ_0

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

Discrete Gaussian Distribution

- Let $D(0, S^T S)$ be discrete Gaussian distribution with 0^n mean and covariance $S^T S$. Then the probability mass function satisfies

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

- Does not satisfy rotational invariance

- Also has a normalizing constant. For now, supported on $\mathbb{Z}^n$

Our Results (Lifting Framework)

Suppose that

- $X \sim D(0, S^T S)$ and $\mathrm{Y} \sim N(0, S^T S)$, 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 $1 - \frac{\delta}{3}$
- $A \in \mathbb{R}^{r \times n}$ 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 estimator h such that $h(A'y) = f(y)$ w.p. $1 - \delta$ for $\mathrm{y} \sim Y + Z$

Example Problem (ℓ_2 Estimation)

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

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

- f satisfies $\Pr_{x \sim X_i, y \sim Y_i}[f(x) \neq f(y)] \leq \frac{\delta}{3}$

Example Problem (ℓ_2 Estimation)

- Suppose there exists a $g(Ax)$ that can distinguish X_1 and X_2

- From our theorem, there exists $h(A'y)$ 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)

- Let $B > 1$ be any fixed desired accuracy parameter.
- Any adversarially robust streaming algorithm which uses a finite-precision linear sketch and B-approximates the ℓ_p norm in a turnstile stream must use $r \geq n - O(\log Bn)$ rows.
- 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.

(Very) High-level Proof Idea

- Essentially, we want to "simulate" continuous Gaussian queries using discrete Gaussian queries.
- Let $\mathcal{D}_{L,S}$ denote the discrete Gaussian distribution on support L and with covariance matrix $S^T S$.
- Let $x \sim \mathcal{D}_{\mathbb{Z}^n, S}$, $y \sim \mathcal{D}_{A\mathbb{Z}^n, SA^T}$
- **As in continuous case, we want to show $\mathbf{d_{TV}(Ax, y)}$ is small on support $\mathbf{A\mathbb{Z}^n}$.**
  - **Lemma [Agarwal-Regev16]: this is true,** under a certain condition for the *orthogonal lattice* to A!

(Very) High-level Idea

1. We design a *pre-processing* for the sketching matrix A, which can be applied without loss of generality, and satisfies the above condition. ➔ ensures that $\mathbf{d_{TV}(Ax, y)}$ is small on support $\mathbf{A\mathbb{Z}^n}$!
2. After applying the pre-processing on sketching matrix A, we show that $Ax + \eta$ and Ag are close in distribution, where η is a uniform noise in the fundamental parallelepiped of the lattice induced by A.
3. WLOG, assume algorithm sees $Ax + \eta$, since algorithm can always round to recover Ax.

Future Directions

Attack ... tches f... n ac... n- de... eams

$r = \Omega(n^{o(1)})$ dimension lower bound for ℓ_0 [GLWYZ24]

→ Attacks on streaming algorithms for ℓ_0 estimation on adversarial insertion-deletion streams

Attac... hes ... d... eams

This work! $r = \Omega(n)$ optimal lower bound for ℓ_p $(p > 0)$

→ Attacks on streaming algorithms for ℓ_p estimation on adversarial insertion-deletion streams

Thank you for listening!