How Robust are Linear Sketches to Adaptive Inputs?

Moritz Hardt, David P. Woodruff

IBM Research Almaden
Two Aspects of Coping with Big Data

**Efficiency:** design algorithms for enormous inputs
- low memory, fast processing time, etc.

**Robustness:** handle adverse conditions
- inputs may be chosen to try to break the algorithm

Can we achieve both?
Algorithmic paradigm: Linear Sketches

Applications: Compressed sensing, data streams, distributed computation, numerical linear algebra

Unifying idea:
Small number of *linear measurements* applied to data

Data vector $x$ in $\mathbb{R}^n$

Sketch $y = Ax$ in $\mathbb{R}^r$

$r << n$
“For each” correctness

For each $x$: \[ \Pr \{ \text{Alg}(x) \text{ correct} \} > 1 - \frac{1}{\text{poly}(n)} \]

$\Pr$ over randomly chosen matrix $A$

Does this imply correctness on many inputs?

Only under \textbf{modeling assumption}: Inputs are non-adaptively chosen

No guarantee if input $x_2$ depends on $\text{Alg}(x_1)$ for earlier input $x_1$

Why not?
Example: Johnson-Lindenstrauss Sketch

- **Goal:** estimate $|x|^2$ from $|Ax|^2$

- **JL Sketch:** if $A$ is a $k \times n$ matrix of i.i.d. $N(0, 1/k)$ random variable with $k > \log n$, then $Pr[|Ax|^2 = (1\pm1/2)|x|^2] > 1-1/poly(n)$

- **Attack:**
  1. Query $x = e_i$ and $x = e_i + e_j$ for all standard unit vectors $e_i$ and $e_j$
     - Learn $|A_i|^2$, $|A_j|^2$, $|A_i + A_j|^2$, so learn $\langle A_i, A_j \rangle$
  2. Hence, learn $A^T A$, and learn kernel of $A$
  3. Query a vector $x$ 2 kernel($A$)
Example: Dynamic Connectivity

- **Goal:** given a dynamic stream of edges to a graph G, find a spanning forest of G

- **Connectivity Sketch [AGM]:** If x is the characteristic vector of edges in \(\{0,1\}^{n^2/2}\), there is a random \(O(n) \times n^2/2\) matrix A with entries in \{-1, 0, 1\} so that from Ax, can recover a spanning forest of G
  - Sketch is correct for poly(n) non-adaptive queries in a stream

- **Attack:**
  1. Let G in G(n,1/2)
  2. Test if edge e in G:
     - Given Ax, delete edges in the spanning forest returned. Repeat until the returned forest is empty or contains e
  3. Can recover G, which has entropy \(n(n-1)/2\). But Ax has entropy \(n \log n\).
Correlations arise in nearly any realistic setting

**Benign/Natural**
Monitor traffic using sketch, re-route traffic based on output, affects future inputs.

**Adversarial**
DoS attack on network monitoring unit

Can we prove correctness?  
Can we thwart the attack?

**In this work:** Strong impossibility results
**Benchmark Problem**

GapNorm($B$): Given $X \in \mathbb{R}^n$ decide if

(YES) $\|X\|_2^2 \geq B$

(NO) $\|X\|_2^2 \leq 1$

**Goal:** Show impossibility for very basic problem.

Easily solvable for $B = 1 + \varepsilon$ using “for each” guarantee by sketch with $O(\log n/\varepsilon^2)$ rows using JL.
Main Result

**Theorem.** For every $B$, given oracle access to a linear sketch using dimension $r \cdot n - \log(Bn)$, we can find in time $\text{poly}(r,B)$ a distribution over inputs on which sketch fails to solve $\text{GapNorm}(B)$.

Efficient attack (rules out crypto), even slightly non-trivial sketching dimension impossible.

**Corollary.** Same result for any $l_p$-norm.

**Corollary.** Same result even if algorithm uses internal randomness on each query.
Application to Compressed Sensing

$l_2/l_2$ recovery: on input $x$, output $x'$ for which:

$$\|x - x'\|_2 \leq C\|x_{\text{tail}(k)}\|_2$$

**Theorem.** No linear sketch with $o(n/C^2)$ rows guarantees $l_2/l_2$ sparse recovery with approximation factor $C$ on a polynomial number of adaptively chosen inputs.

Note: possible with “for each” guarantee with $r = k \log(n/k)$.

[Gilbert-Hemenway-Strauss-W-Wootters12] some positive results
Outline

• Proof of Main Theorem for GapNorm
  – Proved using “Reconstruction Attack”

• Sparse Recovery Result
  – By Reduction from GapNorm
  – Not in this talk
1. Sketches \( Ax \) and \( U^Tx \) are equivalent, where \( U^T \) has orthonormal rows and \( \text{row-span}(U^T) = \text{row-span}(A) \).

2. Sketch \( U^Tx \) equivalent to \( P_U x = UU^T x \)

Why?

Sketch has unbounded computational power on top of \( P_U x \)
Algorithm (Reconstruction Attack)

Input: Oracle access to sketch $f$ using unknown subspace $U$ of dimension $r$

Put $V_0 = \{0\}$, subspace of 0 dimension

For $t = 1$ to $t = r$:

(Correlation Finding) Find vectors $x_1, \ldots, x_m$ weakly correlated with unknown subspace $U$, orthogonal to $V_{t-1}$

(Boosting) Find single vector $x$ strongly correlated with $U$, orthogonal to $V_{t-1}$

(Progress) Put $V_t = \text{span}\{V_{t-1}, x\}$

Output: Subspace $V_r$
Algorithm (Reconstruction Attack)

**Input:** Oracle access to sketch $f$ using unknown subspace $U$ of dimension $r$

Put $V_0 = \{0\}$, empty subspace

For $t = 1$ to $t = r$:

**Correlation Finding** Find vectors $x_1, ..., x_m$ weakly correlated with unknown subspace $U$, orthogonal to $V_{t-1}$

**Boosting** Find single vector $x$ strongly correlated with $U$, orthogonal to $V_{t-1}$

**Progress** Put $V_t = V_{t-1} + \text{span}\{x\}$

**Output:** Subspace $V_r$
Conditional Expectation Lemma

**Lemma.** Given $d$-dimensional sketch $f$, we can find using $\text{poly}(d)$ queries a distribution $g$ such that:

$$
\mathbb{E} \left[ \|P U g\|^2 \mid f(g) = 1 \right] \geq \mathbb{E} \|P U g\|^2 + \Delta
$$

Moreover,

1. $\Delta \geq \text{poly}(1/d)$
2. $g = N(0,\sigma)^n$ for a carefully chosen $\sigma$ unknown to sketching algorithm

“Advantage over random”
Simplification

**Fact:** If \( g \) is Gaussian, then \( P_U g = UU^T g \) is Gaussian as well.

Hence, can think of query distribution as choosing random Gaussian \( g \) to be inside subspace \( U \).

We drop the \( P_U \) projection operator for notational simplicity.
The three step intuition

(Symmetry) Since the queries are random Gaussian inputs $g$ with an unknown variance, by spherical symmetry, sketch $f$ learns nothing more about query distribution than norm $|g|$

(Averaging) If $|g|$ is larger than expected, the sketch is “more likely” to output 1

(Bayes) Hence, by sort-of-Bayes-Rule, conditioned on $f(g)=1$, expectation of $|g|$ is likely to be larger
Def. Let \( p(s) = \Pr\{ f(y) = 1 \} \)
y in U uniformly random with \(|y|^2 = s\)

Fact. If \( g \) is Gaussian with \( \mathbb{E}|g|^2 = t \), then,

\[
\Pr\{f(g) = 1\} = \int_0^\infty p(s)\nu_t(s)ds
\]

density of \( \chi^2 \)-distribution with expectation t
and d degrees of freedom
\[ p(s) = \Pr(f(y) = 1) \]

\[ y \text{ in } U \text{ unif. random with } |y|^2 = s \]

By correctness of sketch

\[ \int_t^\infty p(s) \nu_t(s) ds > \int_0^t p(s) \nu_t(s) ds + \delta \]

\[ \mathbb{E} \left[ \|P_{Ug}\|^2 \mid f(g) = 1 \right] \geq \mathbb{E} \|P_{Ug}\|^2 + \Delta \]
Sliding $\chi^2$-distributions

- $\phi(s) = s_1^B (s-t) v_t(s) \, dt$

If this were instead at least $1/poly(d)$, we’d be done

- $\phi(s) < 0$ unless $s > B - O(B^{1/2} \log B)$
- $s_0^{-1} \phi(s) \, ds = s_0^{-1} s_1^B (s-t) v_t(s) \, dt \, ds = 0$
Averaging Argument

• Recall $p(s) = \Pr[f(y) = 1]$ given uniformly random $|y|^2 = s$

• Correctness:
  – For small $s$, $p(s) \approx 0$, while for large $s$, $p(s) \approx 1$

• $s_0^1 p(s) \phi(s) \, ds$, $d$

• By a calculation, $E[|g_t|^2 \mid f(g_t) = 1]$, $t + \phi$
Algorithm (Reconstruction Attack)

Input: Oracle access to sketch $f$ using unknown subspace $U$ of dimension $r$

Put $V_0 = \{0\}$, empty subspace

For $t = 1$ to $t = r$:

(Correlation Finding) Find vectors $x_1, \ldots, x_m$ weakly correlated with unknown subspace $U$

Boosting) Find single vector $x$ strongly correlated with $U$, orthogonal to $V_{t-1}$

(Progress) Put $V_t = V_{t-1} + \text{span}\{x\}$

Output: Subspace $V_r$
Boosting small correlations

1. Sample $\text{poly}(d)$ vectors using CoEx Lemma

2. Compute top singular vector $x$ of $M$

**Lemma:**

$|P_x x| > 1 - \text{poly}(1/d)$

Proof: Discretization + Concentration

$M = \begin{array}{c}
\text{poly}(d) \\
x_1 \\
x_2 \\
... \\
x_m 
\end{array}$
Implementation in poly(r) time

• W.l.o.g. can assume \( n = r + O(\log nB) \)
  
  – Restrict host space to first \( r + O(\log nB) \) coordinates

• Matrix \( M \) is now \( O(r) \times \text{poly}(r) \)

• Singular vector computation \( \text{poly}(r) \) time
Iterating previous steps

Generalize Gaussian to “subspace Gaussian” = Gaussian vanishing on maintained subspace $V_t$

**Intuition:**

*Each step reduces sketch dimension by one.*

After $r$ steps:
1. Sketch has no dimensions left!
2. Host space still has $n - r > O(\log nB)$ dimensions
Problem

Top singular vector not \textit{exactly} contained in U
Formally, sketch still has dimension $r$

Can fix this by adding small amount of Gaussian noise to all coordinates
Algorithm (Reconstruction Attack)

**Input:** Oracle access to sketch $f$ using unknown subspace $U$ of dimension $r$

Put $V_0 = \{0\}$, empty subspace

**For** $t = 1$ **to** $t = r$:

**(Correlation Finding)** Find vectors $x_1, \ldots, x_m$ weakly correlated with unknown subspace $U$, orthogonal to $V_{t-1}$

**(Boosting)** Find single vector $x$ strongly correlated with $U$, orthogonal to $V_{t-1}$

**(Progress)** Put $V_t = V_{t-1} + \text{span}\{x\}$

**Output:** Subspace $V_r$
Open Problems

• Achievable polynomial dependence still open

• Can efficient linear sketches which tolerate a sufficient polynomial number of adaptive queries be built for interesting problems?

• If you need C adaptive queries, when can you do better than independently repeating the sketch C times?