On Some Open Questions Related to the Log-Approximate-Rank Conjecture

Suhail Sherif

Vector Institute, Toronto

Background: The Log-Approximate-Rank Conjecture

$$\mathrm{R}^{\mathrm{cc}}_{\epsilon}(F)$$

Background: The Log-Approximate-Rank Conjecture

$$\forall F$$

$\log \mathrm{rank}_\epsilon(F)$ $\quad$ $\mathrm{R}^{\mathrm{cc}}_\epsilon(F)$ $\quad$ $\mathrm{rank}_\epsilon(F)$

Background: The Log-Approximate-Rank Conjecture

$$\forall F$$

$\log \mathrm{rank}_\epsilon(F)$ $\quad$ $\mathrm{R}^{\mathrm{cc}}_\epsilon(F)$ $\quad$ $\mathrm{rank}_\epsilon(F)$

Communication protocol for F of cost $k \implies$ Rank-2^k decomposition of matrix approximating F.
[Krause '96]

Background: The Log-Approximate-Rank Conjecture

$\forall F$

$\log \mathrm{rank}_\epsilon(F)$ $\qquad$ $\mathrm{R}^{\mathrm{cc}}_\epsilon(F)$ $\qquad$ $\mathrm{rank}_\epsilon(F)$

Communication protocol for F of cost $k \implies$ Rank-2^k decomposition of matrix approximating F. [Krause '96]

There is a communication protocol for F of cost $O(\mathrm{rank}_\epsilon(F))$. [Gál and Syed '19]

The Log-Approximate-Rank Conjecture

$$\forall F$$

$\log \operatorname{rank}_\epsilon(F)$ $\quad$ $\mathrm{R}^{\mathrm{cc}}_\epsilon(F)$ $\quad$ $\operatorname{rank}_\epsilon(F)$

Refuting The Log-Approximate-Rank Conjecture

[Chattopadhyay Mande S ‘19]

$\exists F$

$\log \mathrm{rank}_\epsilon(F)$ $\mathrm{R}^{\mathrm{cc}}_\epsilon(F)$ $\mathrm{rank}_\epsilon(F)$

Refuting The Log-Approximate-Rank Conjecture

[Chattopadhyay Mande S '19]

$\exists F$

$\log \text{rank}_\epsilon(F)$ $\text{R}^{\text{cc}}_\epsilon(F)$ $\text{rank}_\epsilon(F)$

Showed F such that $\text{R}^{\text{cc}}_\epsilon(F) \geq \sqrt[4]{\text{rank}_\epsilon(F)}$.

Refuting The Log-Approximate-Rank Conjecture

[Chattopadhyay Mande S '19]

$\exists F$

$\log \operatorname{rank}_\epsilon(F)$ $\mathrm{R}^{cc}_\epsilon(F)$ $\operatorname{rank}_\epsilon(F)$

Showed F such that $\mathrm{R}^{cc}_\epsilon(F) \geq \sqrt[4]{\operatorname{rank}_\epsilon(F)}$.

1. Can we get closer?

Refuting The Log-Approximate-Rank Conjecture

[Chattopadhyay Mande S '19]

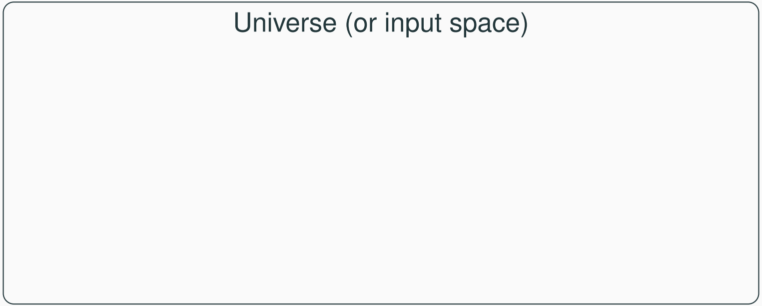
$\exists F$

$\log \operatorname{rank}_\epsilon(F)$ $\mathrm{R}^{\mathrm{cc}}_\epsilon(F)$ $\operatorname{rank}_\epsilon(F)$

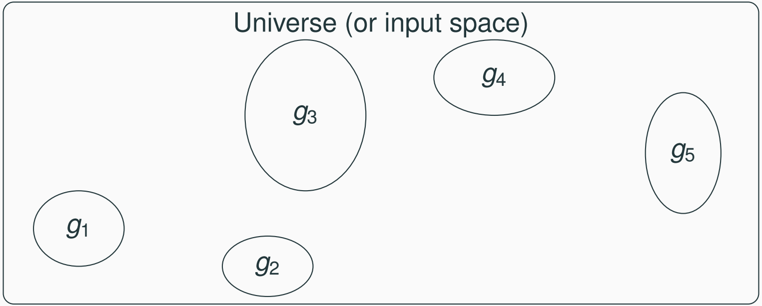
Showed F such that $\mathrm{R}^{\mathrm{cc}}_\epsilon(F) \geq \sqrt[4]{\operatorname{rank}_\epsilon(F)}$.

1. Can we get closer?
2. Can we refute $\mathrm{R}^{\mathrm{cc}}_\epsilon(F) \leq \log\left(\max\{\operatorname{rank}^+_\epsilon(F), \operatorname{rank}^+_\epsilon(\neg F)\}\right)^{O(1)}$?
[Kol Moran Shpilka Yehudayoff '14]

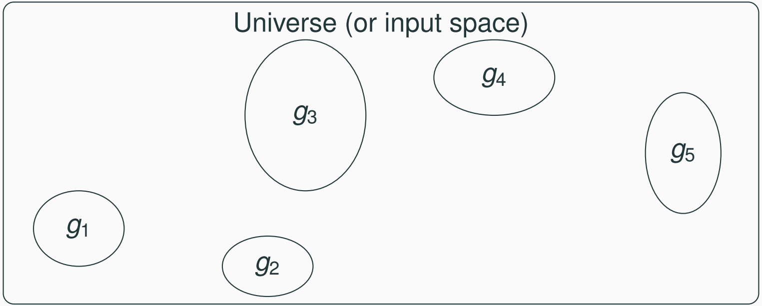
Functions with small Approximate Rank



Functions with small Approximate Rank

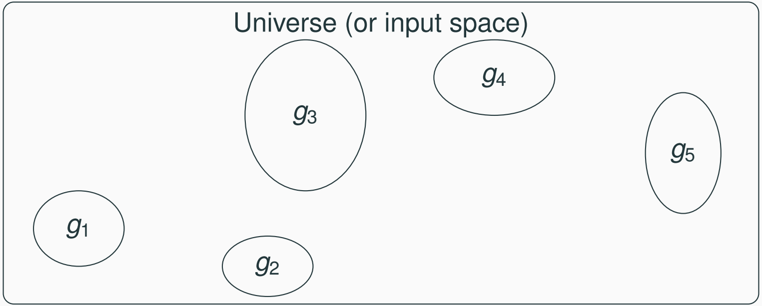


Functions with small Approximate Rank



$$f(x) := \bigvee_{i\in[t]} g_i(x)$$

Functions with small Approximate Rank



$$f(x) := \bigvee_{i\in[t]} g_i(x)$$

$$f(x) := \sum_{i\in[t]} g_i(x)$$

Functions with small Approximate Rank

Rank is subadditive, so

approximate rank of f is $\approx$ approximate rank of g_i times t.

$$f(x) := \bigvee_{i \in [t]} g_i(x)$$

$$f(x) := \sum_{i \in [t]} g_i(x)$$

Functions with small Approximate Rank

Rank is subadditive, so
approximate rank of f is $\approx$ approximate rank of g_i times t.

Log-Approximate-Rank Conjecture implies cost of computing f is $\mathrm{poly}(\log t,$ cost of computing $g)$.

$$f(x) := \bigvee_{i \in [t]} g_i(x)$$

$$f(x) := \sum_{i \in [t]} g_i(x)$$

Functions with small Approximate Rank

Rank is subadditive, so
approximate rank of f is $\approx$ approximate rank of g_i times t.

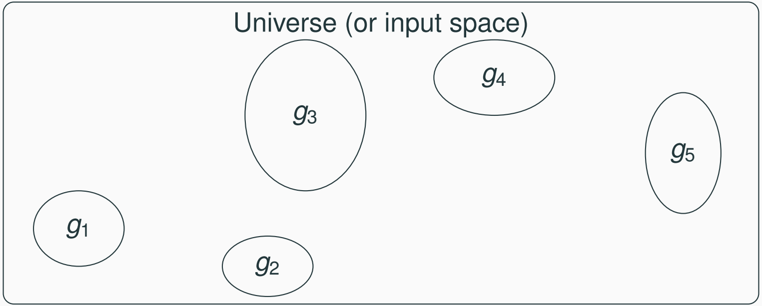
Log-Approximate-Rank Conjecture implies cost of computing f is
$\mathrm{poly}(\log t,$ cost of computing $g)$.

Can't we be forced to compute each g_i, resulting in $\Omega(t)$ cost?

$$f(x) := \bigvee_{i \in [t]} g_i(x)$$

$$f(x) := \sum_{i \in [t]} g_i(x)$$

Functions with small Approximate Rank



$$f(x) := \bigvee_{i\in[t]} g_i(x)$$

$$f(x) := \sum_{i\in[t]} g_i(x)$$

Showing Hardness

1. Work with functions $f : \{0,1\}^n \to \{0,1\}$ that are sums of simple functions (subcube/subspace indicators).

Showing Hardness

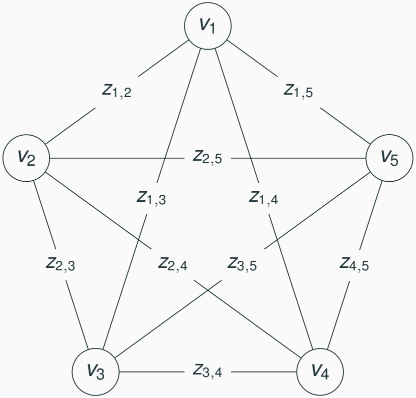
1. Work with functions $f : \{0,1\}^n \to \{0,1\}$ that are sums of simple functions (subcube/subspace indicators).
2. Sanity check: Show that these are hard for randomized parity decision trees. (Can make linear queries of the form $\langle v, x\rangle$ thinking of $x \in \mathbb{F}_2^n$.)

Showing Hardness

1. Work with functions $f : \{0,1\}^n \to \{0,1\}$ that are sums of simple functions (subcube/subspace indicators).
2. Sanity check: Show that these are hard for randomized parity decision trees. (Can make linear queries of the form $\langle v, x\rangle$ thinking of $x \in \mathbb{F}_2^n$.)
3. Show that $f \circ \mathrm{XOR}$ is hard for randomized communication protocols.

The counterexample

$$\text{SINK} : \{0,1\}^{\binom{m}{2}} \to \{0,1\}$$

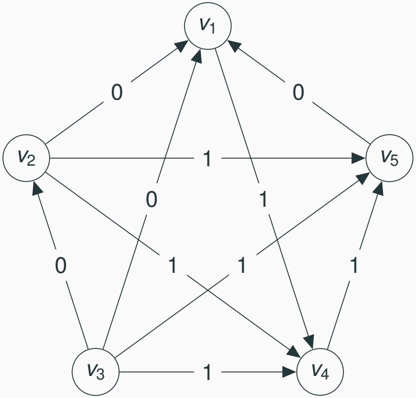


The input bits of SINK orient the edges of the complete graph.

$\text{SINK}(z) = 1$ iff there is a sink in the directed graph G_z.

The counterexample

$$\text{SINK} : \{0,1\}^{\binom{m}{2}} \to \{0,1\}$$

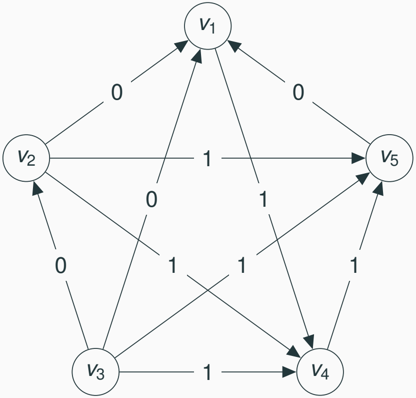


The input bits of SINK orient the edges of the complete graph.

SINK(z) $= 1$ iff there is a sink in the directed graph G_z.

The counterexample

$$\text{SINK} : \{0,1\}^{\binom{m}{2}} \to \{0,1\}$$



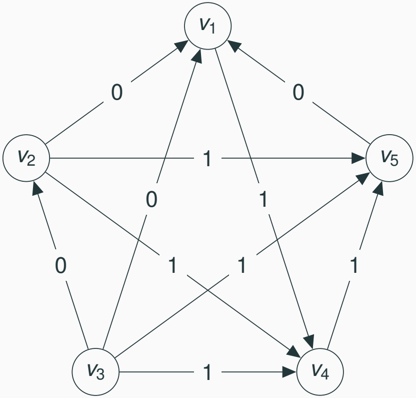
The input bits of SINK orient the edges of the complete graph.

SINK$(z) = 1$ iff there is a sink in the directed graph G_z.

SINK $= \sum_{i \in [n]}$ SINK$_i$. Each SINK$_i$ is a subcube.

The counterexample

$$\mathrm{SINK} : \{0,1\}^{\binom{m}{2}} \to \{0,1\}$$



A subspace/rectangle A that is biased against v_i being a sink must have a slightly small $A|_i$.

A subspace/rectangle A that is biased against inputs with sinks must be slightly small for many $A|_i$s.

A subspace/rectangle A that is biased against inputs with sinks must be very small. (Shearer's Lemma)

Doing Better Than $\sqrt[4]{\mathrm{rank}_\epsilon(F)}$

with Arkadev Chattopadhyay and Ankit Garg

Dual Subspace Designs

Subspace Designs [Guruswami Xing ‘12]:

A set of subspaces such that any small subspace intersects only a few of them non-trivially.

$S_1, S_2, \ldots, S_k \subset \mathbb{F}_2^n$ such that
$\forall W \subset \mathbb{F}_2^n$ with $\dim(W) < a$,
$W \cap S_i \neq \{0\}$ for only h values of i.

Dual Subspace Designs

Subspace Designs [Guruswami Xing '12]:

A set of subspaces such that any small subspace intersects only a few of them non-trivially.

Taking the duals:

A set of subspaces such that any large subspace can be biased against only a few them.

$S_1, S_2, \ldots, S_k \subset \mathbb{F}_2^n$ such that
$\forall W \subset \mathbb{F}_2^n$ with $\mathrm{codim}(W) < a$,
$\frac{|W \cap S_i|}{|W|} \neq \frac{|S_i|}{2^n}$ for only h values of i.

Dual Subspace Designs

Subspace Designs [Guruswami Xing ‘12]:

A set of subspaces such that any small subspace intersects only a few of them non-trivially.

Taking the duals:

A set of subspaces such that any large subspace can be biased against only a few them.

Randomized Parity Decision Tree lower bound follows immediately.

Open Question 1

Can we prove a communication lower bound of $\Omega(n)$ against (union of a dual subspace design)$\circ$XOR?

Open Question 1

Can we prove a communication lower bound of $\Omega(n)$ against (union of a dual subspace design)$\circ$XOR?

Efficient subspace designs are known to exist.

Open Question 1

Can we prove a communication lower bound of $\Omega(n)$ against (union of a dual subspace design)$\circ$XOR?

Efficient subspace designs are known to exist.

Would give F such that $\mathrm{R}^{\mathrm{cc}}_{\epsilon}(F) \geq \sqrt[3]{\mathrm{rank}_{\epsilon}(F)}$.

Open Question 1

Can we prove a communication lower bound of $\Omega(n)$ against (union of a dual subspace design)$\circ$XOR?

Efficient subspace designs are known to exist.

Would give F such that $\mathrm{R}^{\mathrm{cc}}_{\epsilon}(F) \geq \sqrt[3]{\mathrm{rank}_{\epsilon}(F)}$.

Do subspace designs have an analog of Shearer's lemma?

Open Question 1.1

Our Conjecture

If a distribution A over $\{0,1\}^n$ satisfies $H(A|_T) \leq \mathrm{codim}(T) - \Omega(1)$
for many subspaces T from a subspace design,
then $H(A) < n - \Omega(n)$.

A concrete problem: CyclicShift

$$f : \{0,1\}^{n+n} \to \{0,1\}$$

$$f : (x, y) \mapsto 1 \text{ iff } \exists i \in [n] \text{ such that } x_{\to i} = y$$

$$\left((a_1 a_2 a_3 a_4 a_5)_{\to 2} = (a_4 a_5 a_1 a_2 a_3) \right)$$

A concrete problem: CyclicShift

$$f : \{0,1\}^{n+n} \to \{0,1\}$$

$$f : (x,y) \mapsto 1 \text{ iff } \exists i \in [n] \text{ such that } x_{\to i} = y$$

$$\left((a_1 a_2 a_3 a_4 a_5)_{\to 2} = (a_4 a_5 a_1 a_2 a_3) \right)$$

Even the randomized parity decision tree complexity of CyclicShift is unknown.

A concrete problem: CyclicShift

$$f : \{0,1\}^{n+n} \to \{0,1\}$$

$$f : (x, y) \mapsto 1 \text{ iff } \exists i \in [n] \text{ such that } x_{\to i} = y$$

$$\Big((a_1 a_2 a_3 a_4 a_5)_{\to 2} = (a_4 a_5 a_1 a_2 a_3) \Big)$$

Even the randomized parity decision tree complexity of CyclicShift is unknown.

(Function inspired by a function from 'String Matching: Communication, Circuits, and Learning', by Golovnev, Göös, Reichman and Shinkar '19)

Approximate Nonnegative Rank

with Arkadev Chattopadhyay

The issue

The function SINK was a sum of simple functions.

¬SINK was not.

Rectifying it in 3 simple steps

- Partition $\{0,1\}^n$ into two sets so that both parts are unions of a few disjoint subcubes.

Rectifying it in 3 simple steps

- Partition $\{0, 1\}^n$ into two sets so that both parts are unions of a few disjoint subcubes.
- Let f evaluate to 1 on one of the parts and 0 on the other part.

Rectifying it in 3 simple steps

- Partition $\{0,1\}^n$ into two sets so that both parts are unions of a few disjoint subcubes.
- Let f evaluate to 1 on one of the parts and 0 on the other part.
- Realize that any such f has small randomized parity decision tree complexity.

Rectifying it in 3 simple steps

- Partition $\{0, 1\}^n$ into two sets so that both parts are unions of a few disjoint subcubes.
- Let f evaluate to 1 on one of the parts and 0 on the other part.
- Realize that any such f has small randomized parity decision tree complexity.

If $f^{-1}(0)$ and $f^{-1}(1)$ can be covered by c monochromatic subcubes, there is a size-$2^{\mathrm{polylog}(c,n)}$ decision tree computing f. [Ehrenfeucht and Haussler ‘89]

Can subspaces help here?

Can we partition $\{0, 1\}^n$ into two sets so that both parts are unions of few disjoint subspaces, while the resulting f remains hard for randomized parity decision trees?

Can subspaces help here?

Can we partition $\{0,1\}^n$ into two sets so that both parts are unions of few disjoint subspaces, while the resulting f remains hard for randomized parity decision trees?

Negation:
For any partition of $\{0,1\}^n$ into subspaces $A_1, A_2, \ldots, A_k$, there is an efficient randomized parity decision tree that computes $x \mapsto i$ s.t. $x \in A_i$.

Can subspaces help here?

Can we partition $\{0,1\}^n$ into two sets so that both parts are unions of few disjoint subspaces, while the resulting f remains hard for randomized parity decision trees?

Negation:
For any partition of $\{0,1\}^n$ into subspaces $A_1, A_2, \ldots, A_k$, there is an efficient randomized parity decision tree that computes $x \mapsto i$ s.t. $x \in A_i$.

Conjecture:
For any partition of $\{0,1\}^n$ into subspaces, there is a tree-like partition of $\{0,1\}^n$ that refines it without having too many more parts.

Thank you. ~~I am now open to questions.~~ The questions are now open to you.