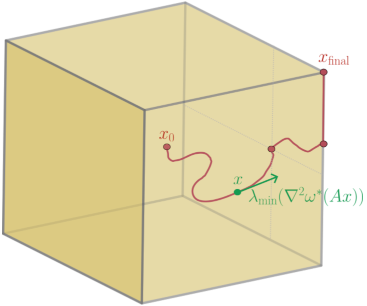
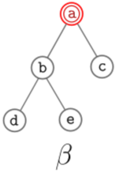
Up to the constant: discrepancy theory and iterative algorithms on random matrices



$$Z_a^\alpha = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

$$Z_a^\beta = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



Chris Jones

Bocconi University

Lucas Pesenti

Bocconi University

Adrian Vladu

CNRS & IRIF
Université Paris Cité

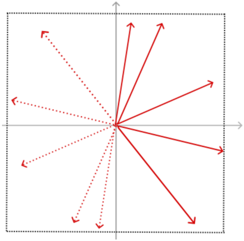
Discrepancy Theory (1/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$

Discrepancy Theory (1/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$
achieving small discrepancy:

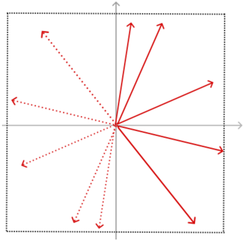
$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$



Discrepancy Theory (1/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$
achieving small discrepancy:

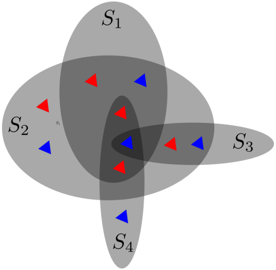
$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$



Ex: set system $S_1, \ldots, S_d$ over n elements

$$u_{i,j} = \begin{cases} 1 & \text{if } i \in S_j \\ 0 & \text{otherwise} \end{cases}$$

Goal: red/blue coloring with small maximal imbalance

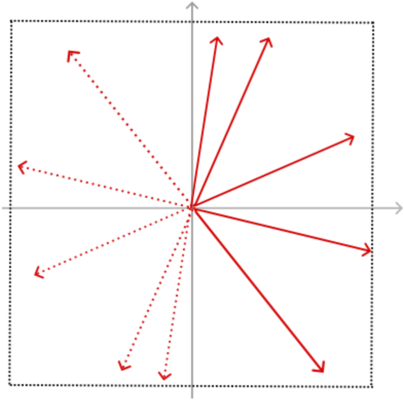


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

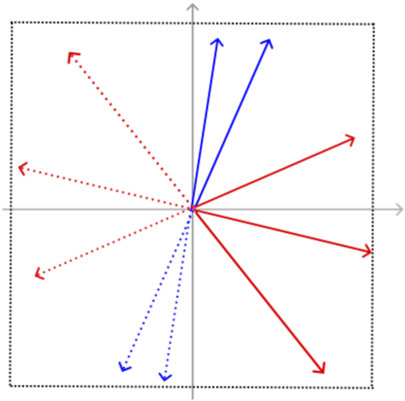


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

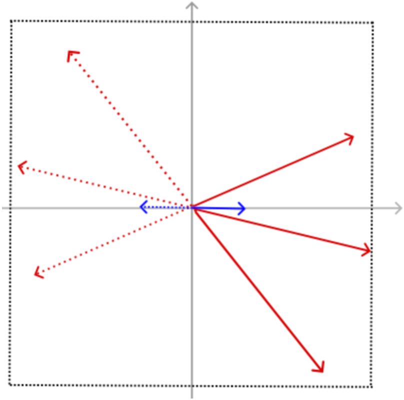


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

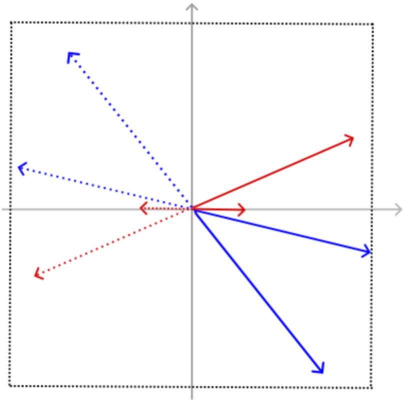


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

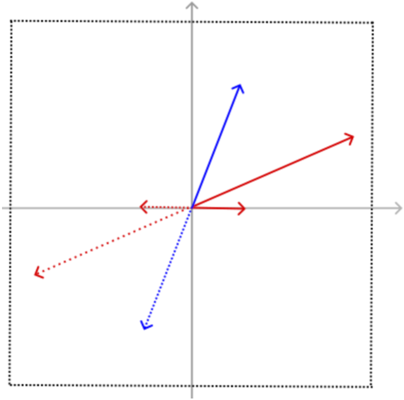


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

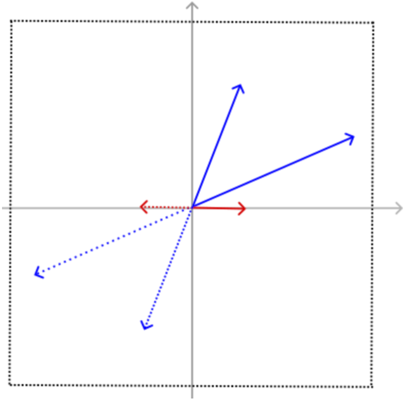


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

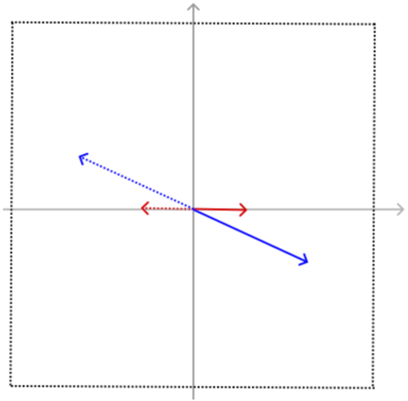


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

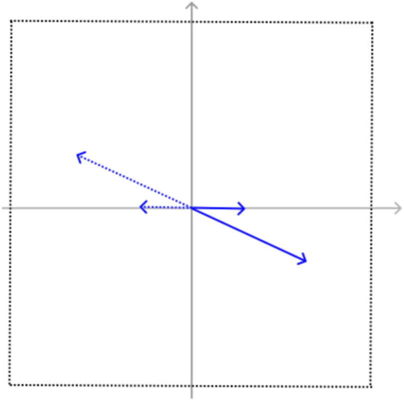


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

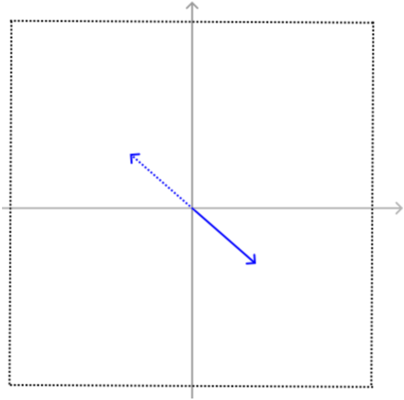


Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$



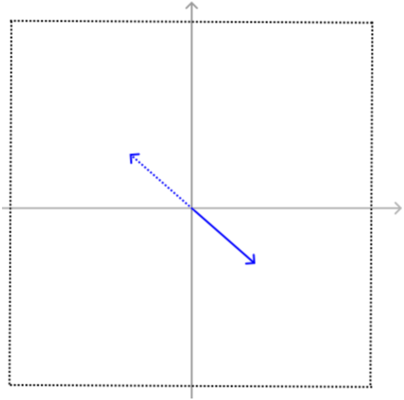
Discrepancy Theory (2/2)

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Ex: $d = 2$

There exists a coloring with discrepancy $\leqslant 2$



Spencer's Theorem

High-dimensional regime: $d = n$

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

Spencer's Theorem

High-dimensional regime: $d = n$

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

1. Sample i.i.d. $x_1, \ldots, x_n \sim \{-1, 1\}$

Spencer's Theorem

High-dimensional regime: $d = n$

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

1. Sample i.i.d. $x_1, \ldots, x_n \sim \{-1, 1\}$
2. Each coordinate of $\sum_i x_i \mathbf{u}_i$ is $\lesssim \mathcal{N}(0, n)$

Spencer's Theorem

High-dimensional regime: $d = n$

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

1. Sample i.i.d. $x_1, \ldots, x_n \sim \{-1, 1\}$
2. Each coordinate of $\sum_i x_i \mathbf{u}_i$ is $\lesssim \mathcal{N}(0, n)$
3. Union bound: w.h.p. they are all $O(\sqrt{n \log n})$

Spencer's Theorem

High-dimensional regime: $d = n$

Input: $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \ldots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

1. Sample i.i.d. $x_1, \ldots, x_n \sim \{-1, 1\}$
2. Each coordinate of $\sum_i x_i \mathbf{u}_i$ is $\lesssim \mathcal{N}(0, n)$
3. Union bound: w.h.p. they are all $O(\sqrt{n \log n})$

Thm: [Spencer'85] there always exist colorings of discrepancy $\leqslant 6\sqrt{n}$

Spencer's Theorem

High-dimensional regime: $d = n$

Input: $\mathbf{u}_1, \dots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$
Output: a coloring $\mathbf{x} = (x_1, \dots, x_n) \in \{-1, 1\}^n$ achieving small discrepancy:

$$\text{discrepancy}(\mathbf{x}) := \left\| \sum_{i=1}^{n} x_i \mathbf{u}_i \right\|_\infty$$

1. Sample i.i.d. $x_1, \dots, x_n \sim \{-1, 1\}$
2. Each coordinate of $\sum_i x_i \mathbf{u}_i$ is $\lesssim \mathcal{N}(0, n)$
3. Union bound: w.h.p. they are all $O(\sqrt{n \log n})$

Thm: [Spencer'85] there always exist colorings of discrepancy $\leqslant 6\sqrt{n}$

Today: algorithmic proof of this theorem improving the 6

Sparsification via Discrepancy

Discrepancy: given $\mathcal{O}_1, \ldots, \mathcal{O}_n$, find $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\left\| \sum_i x_i \mathcal{O}_i \right\|$ is small

$\longrightarrow$

Sparsification: given $\mathcal{O}_1, \ldots, \mathcal{O}_n$, find $S \subseteq [n]$ s.t. $|S| \ll n$ and $\sum_{i \in S} \mathcal{O}_i \approx \sum_{i \in [n]} \mathcal{O}_i$

Sparsification via Discrepancy

Discrepancy: given $\mathcal{O}_1, \ldots, \mathcal{O}_n$, find $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\left\| \sum_i x_i \mathcal{O}_i \right\|$ is small

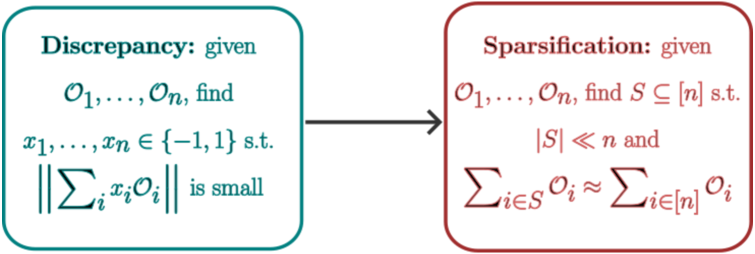
→

Sparsification: given $\mathcal{O}_1, \ldots, \mathcal{O}_n$, find $S \subseteq [n]$ s.t. $|S| \ll n$ and $\sum_{i \in S} \mathcal{O}_i \approx \sum_{i \in [n]} \mathcal{O}_i$

Why sparsification?

- fast algorithms
- low memory

Sparsification via Discrepancy



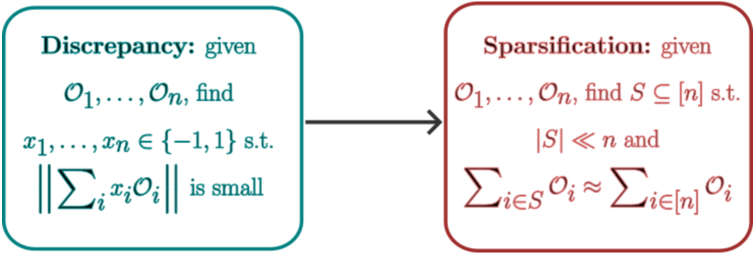
Why sparsification?

- fast algorithms
- low memory

Reduction:

1. Solve the discrepancy problem to get $\mathbf{x}$
2. $\sum_i (1 \pm x_i)\mathcal{O}_i \approx \sum_i \mathcal{O}_i$ and has support $\leqslant n/2$
3. Repeat!

Sparsification via Discrepancy



Why sparsification?

- fast algorithms
- low memory

Reduction:

1. Solve the discrepancy problem to get $\mathbf{x}$
2. $\sum_i (1 \pm x_i)\mathcal{O}_i \approx \sum_i \mathcal{O}_i$ and has support $\leqslant n/2$
3. Repeat!

Examples:

- [Reis–Rothvoss'20; Jambulapati–Reis–Tian'24] sparsify graphs
- [Reis–Rothvoss'22] sparsify convex combinations
- [Bozzai–Reis–Rothvoss'23] sparsify zonotopes

Discrepancy and Continuous Methods

Thm: [Spencer'85] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty = O(\sqrt{n})$

Discrepancy and Continuous Methods

Thm: [Spencer'85] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty = O(\sqrt{n})$

Already different known *algorithmic* proofs:

- [Bansal'10, Lovett–Meka'12] random walks, SDP
- [Eldan–Singh'14, Rothvoss'14] LP with random objective
- [Levy–Ramadas–Rothvoss'17] multiplicative weights update
- [Bansal–Laddha–Vempala'22] barrier potential function

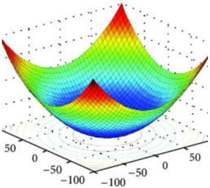
Discrepancy and Continuous Methods

Thm: [Spencer'85] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty = O(\sqrt{n})$

Already different known *algorithmic* proofs:

- [Bansal'10, Lovett–Meka'12] random walks, SDP
- [Eldan–Singh'14, Rothvoss'14] LP with random objective
- [Levy–Ramadas–Rothvoss'17] multiplicative weights update
- [Bansal–Laddha–Vempala'22] barrier potential function

More and more inspired by continuous optimization



Today: Newton's method on a regularized objective

Spencer's Theorem via Regularization (1/3)

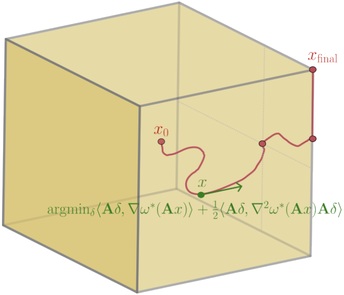
Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|A_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{Ax}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $\mathbf{x} \mapsto \omega^*(\mathbf{Ax})$

Spencer's Theorem via Regularization (1/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n\times n}$ s.t. $|A_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{Ax}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $\mathbf{x} \mapsto \omega^*(\mathbf{Ax})$



Algorithm:

Start from $\mathbf{x} = (0, \ldots, 0)$

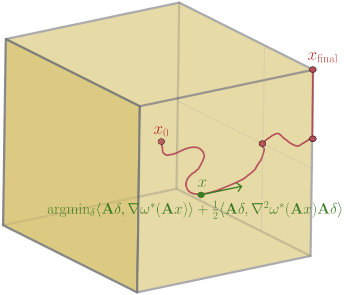
While $\mathbf{x} \notin \{\pm 1\}^n$

$F := \{i : x_i \notin \{-1, 1\}\}$

Spencer's Theorem via Regularization (1/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|A_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{Ax}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $\mathbf{x} \mapsto \omega^*(\mathbf{Ax})$



Algorithm:

Start from $\mathbf{x} = (0, \ldots, 0)$

While $\mathbf{x} \notin \{\pm 1\}^n$

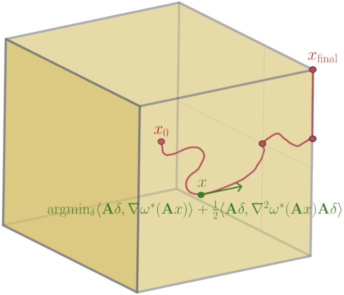
$F := \{i : x_i \notin \{-1, 1\}\}$

Find δ s.t. $\text{supp}(\delta) \subseteq F$, $\delta \perp \mathbf{x}$ minimizing
$\langle \mathbf{A}\delta, \nabla\omega^*(\mathbf{Ax})\rangle + \frac{1}{2}\langle \mathbf{A}\delta, \nabla^2\omega^*(\mathbf{Ax})\mathbf{A}\delta\rangle$

Spencer's Theorem via Regularization (1/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|A_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{Ax}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $\mathbf{x} \mapsto \omega^*(\mathbf{Ax})$



Algorithm:

Start from $\mathbf{x} = (0, \ldots, 0)$

While $\mathbf{x} \notin \{\pm 1\}^n$

- $F := \{i : x_i \notin \{-1, 1\}\}$
- Find δ s.t. $\text{supp}(\delta) \subseteq F$, $\delta \perp \mathbf{x}$ minimizing $\langle \mathbf{A}\delta, \nabla\omega^*(\mathbf{Ax})\rangle + \frac{1}{2}\langle \mathbf{A}\delta, \nabla^2\omega^*(\mathbf{Ax})\mathbf{A}\delta\rangle$
- $\mathbf{x}$ makes a small step in direction δ while staying in $[-1, 1]^n$

Spencer's Theorem via Regularization (2/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}\mathbf{x}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Spencer's Theorem via Regularization (2/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}\mathbf{x}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Up to tracking $\begin{bmatrix} +\mathbf{A} & \mathbf{0} \\ -\mathbf{A} & \mathbf{0} \end{bmatrix}$, assume WLOG

$$\|\mathbf{A}\mathbf{x}\|_\infty = \max_{1 \leqslant i \leqslant n} (\mathbf{A}\mathbf{x})_i = \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{A}\mathbf{x}, \mathbf{r} \rangle$$

where $\Delta_n := \{\mathbf{r} \in \mathbb{R}^n : r_i \geqslant 0, \sum_i r_i = 1\}$.

Spencer's Theorem via Regularization (2/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}\mathbf{x}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Up to tracking $\begin{bmatrix} +\mathbf{A} & \mathbf{0} \\ -\mathbf{A} & \mathbf{0} \end{bmatrix}$, assume WLOG

$$\|\mathbf{A}\mathbf{x}\|_\infty = \max_{1 \leqslant i \leqslant n} (\mathbf{A}\mathbf{x})_i = \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{A}\mathbf{x}, \mathbf{r} \rangle$$

where $\Delta_n := \{\mathbf{r} \in \mathbb{R}^n : r_i \geqslant 0, \sum_i r_i = 1\}$.

Def: Regularized maximum

$$\omega^*(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle + \sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{n}} \mathbf{r}_{\mathbf{i}}^{\frac{1}{2}} .$$

Spencer's Theorem via Regularization (2/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}\mathbf{x}\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Up to tracking $\begin{bmatrix} +\mathbf{A} & \mathbf{0} \\ -\mathbf{A} & \mathbf{0} \end{bmatrix}$, assume WLOG

$$\|\mathbf{A}\mathbf{x}\|_\infty = \max_{1 \leqslant i \leqslant n} (\mathbf{A}\mathbf{x})_i = \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{A}\mathbf{x}, \mathbf{r} \rangle$$

where $\Delta_n := \{\mathbf{r} \in \mathbb{R}^n : r_i \geqslant 0, \sum_i r_i = 1\}$.

Def: Regularized maximum

$$\omega^*(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle + \sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{n}} \mathbf{r}_{\mathbf{i}}^{\frac{1}{2}} .$$

Claim: $\omega^*(\mathbf{z}) = \|\mathbf{z}\|_\infty \pm O(\sqrt{n})$

Spencer's Theorem via Regularization (3/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}x\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Analysis:

Spencer's Theorem via Regularization (3/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n\times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}x\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Analysis:

1. By bounding $\nabla^2\omega^*(\mathbf{Ax})$, we prove that there always exists a direction $\delta \perp \mathbf{x}$, $\mathrm{supp}(\delta) \subseteq F$ s.t.

$$\omega^*(\mathbf{A}(\mathbf{x}+\delta)) - \omega^*(\mathbf{Ax}) \leqslant \frac{\|\delta\|_2^2}{\sqrt{|F|}}\,.$$

Spencer's Theorem via Regularization (3/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}x\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Analysis:

1. By bounding $\nabla^2 \omega^*(\mathbf{Ax})$, we prove that there always exists a direction $\delta \perp \mathbf{x}$, $\mathrm{supp}(\delta) \subseteq F$ s.t.

$$\omega^*(\mathbf{A}(\mathbf{x}+\delta)) - \omega^*(\mathbf{Ax}) \leqslant \frac{\|\delta\|_2^2}{\sqrt{|F|}}\,.$$

2. Hence we get charged $\frac{1}{\sqrt{|F|}}$ discrepancy "per unit of $\|\mathbf{x}\|_2^2$"

Spencer's Theorem via Regularization (3/3)

Thm: [Spencer'85] For any $\mathbf{A} \in \mathbb{R}^{n \times n}$ s.t. $|\mathbf{A}_{ij}| \leqslant 1$, there exists $\mathbf{x} \in \{\pm 1\}^n$ s.t. $\|\mathbf{A}x\|_\infty = O(\sqrt{n})$

1. Define a "smooth" proxy $\omega^*(\cdot)$ for $\|\cdot\|_\infty$
2. Run "sticky" Newton's Method on $x \mapsto \omega^*(\mathbf{A}x)$

Analysis:

1. By bounding $\nabla^2 \omega^*(\mathbf{Ax})$, we prove that there always exists a direction $\delta \perp \mathbf{x}$, $\mathrm{supp}(\delta) \subseteq F$ s.t.
$$\omega^*(\mathbf{A}(\mathbf{x} + \delta)) - \omega^*(\mathbf{Ax}) \leqslant \frac{\|\delta\|_2^2}{\sqrt{|F|}}.$$
2. Hence we get charged $\frac{1}{\sqrt{|F|}}$ discrepancy "per unit of $\|\mathbf{x}\|_2^2$"
3. Worst case: one coordinate of $\mathbf{x}$ get frozen every time $\|\mathbf{x}\|_2^2$ increases by 1, total cost
$$1 \times \frac{1}{\sqrt{n}} + 1 \times \frac{1}{\sqrt{n-1}} + \ldots = O(\sqrt{n}). \quad \square$$

The power of regularization

Thm: [P.–Vladu'23] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty \leqslant 3.8\sqrt{n}$

Open question 1: What is the best possible constant?

The power of regularization

Thm: [P.–Vladu'23] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty \leqslant 3.8\sqrt{n}$

Open question 1: What is the best possible constant?

Open question 2: What are other regularizers for this problem?

The power of regularization

Thm: [P.–Vladu'23] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty \leqslant 3.8\sqrt{n}$

Open question 1: What is the best possible constant?

Open question 2: What are other regularizers for this problem?

$$\omega^*(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle + \sum_{i=1}^{n} r_i^{\frac{1}{2}} = \min_{\lambda \geqslant \max z_i} \lambda + \sum_{i=1}^{n} \frac{1}{\lambda - z_i}$$

[Audibert–Bubeck'09; Allen-Zhu–Liao–Orecchia'15]

The power of regularization

Thm: [P.–Vladu'23] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty \leqslant 3.8\sqrt{n}$

Open question 1: What is the best possible constant?

Open question 2: What are other regularizers for this problem?

$$\omega^*(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle + \sum_{i=1}^n r_i^{\frac{1}{2}} = \min_{\lambda \geqslant \max z_i} \lambda + \sum_{i=1}^n \frac{1}{\lambda - z_i}$$

[Audibert–Bubeck'09; Allen-Zhu–Liao–Orecchia'15]

Ex: What about

$$\operatorname{smax}(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle - \sum_{i=1}^n r_i \log r_i = \log \left(\sum_{i=1}^n e^{z_i} \right) ?$$

The power of regularization

Thm: [P.–Vladu'23] For any $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^n$ s.t. $\|\mathbf{u}_i\|_\infty \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{u}_i\|_\infty \leqslant 3.8\sqrt{n}$

Open question 1: What is the best possible constant?

Open question 2: What are other regularizers for this problem?

$$\omega^*(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle + \sum_{i=1}^n r_i^{\frac{1}{2}} = \min_{\lambda \geqslant \max z_i} \lambda + \sum_{i=1}^n \frac{1}{\lambda - z_i}$$

[Audibert–Bubeck'09; Allen-Zhu–Liao–Orecchia'15]

Ex: What about

$$\mathrm{smax}(\mathbf{z}) := \max_{\mathbf{r} \in \Delta_n} \langle \mathbf{z}, \mathbf{r} \rangle - \sum_{i=1}^n r_i \log r_i = \log\left(\sum_{i=1} e^{z_i}\right)?$$

Regularized Newton's method with smax ≈ multiplicative weights update ≈ derandomizing the coin flipping argument

Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{R}^{n \times n}$ s.t. $\|\mathbf{A}_i\|_{\mathrm{op}} \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{\mathrm{op}} = O(\sqrt{n})$

Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathbb{R}^{n\times n}$ s.t. $\|\mathbf{A}_i\|_{\mathrm{op}} \leqslant 1$, there exists $x_1, \dots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{\mathrm{op}} = O(\sqrt{n})$

Random coloring: $\sqrt{\log n} \cdot \left\|\sum_i \mathbf{A}_i^2\right\|_{\mathrm{op}}^{\frac{1}{2}} \leqslant \sqrt{n \log n}$

Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{R}^{n \times n}$ s.t. $\|\mathbf{A}_i\|_{\mathrm{op}} \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{\mathrm{op}} = O(\sqrt{n})$

Random coloring: $\sqrt{\log n} \cdot \left\|\sum_i \mathbf{A}_i^2\right\|_{\mathrm{op}}^{\frac{1}{2}} \leqslant \sqrt{n \log n}$

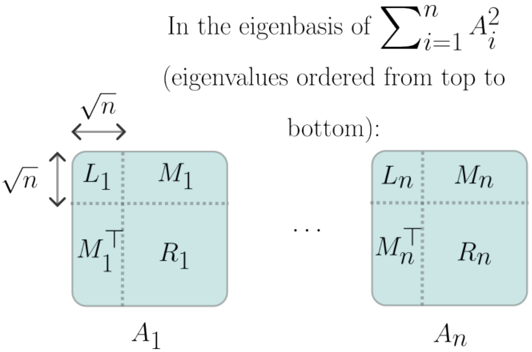
Thm: [Hopkins–Raghavendra–Shetty'22] conjecture holds if $\mathrm{rank}(\mathbf{A}_i) \ll \sqrt{n}$

Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{R}^{n \times n}$ s.t. $\|\mathbf{A}_i\|_{\text{op}} \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{\text{op}} = O(\sqrt{n})$

Random coloring: $\sqrt{\log n} \cdot \|\sum_i \mathbf{A}_i^2\|_{\text{op}}^{\frac{1}{2}} \leqslant \sqrt{n \log n}$

Thm: [Hopkins–Raghavendra–Shetty'22] conjecture holds if $\text{rank}(\mathbf{A}_i) \ll \sqrt{n}$

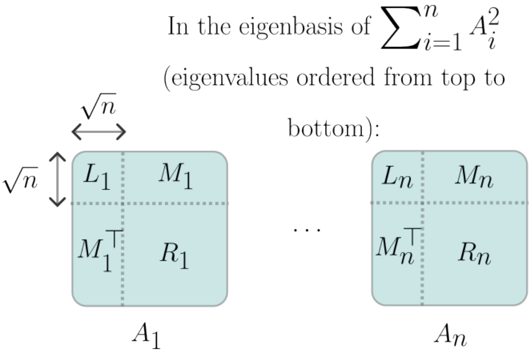


Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{R}^{n \times n}$ s.t. $\|\mathbf{A}_i\|_{\mathrm{op}} \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{\mathrm{op}} = O(\sqrt{n})$

Random coloring: $\sqrt{\log n} \cdot \|\sum_i \mathbf{A}_i^2\|_{\mathrm{op}}^{\frac{1}{2}} \leqslant \sqrt{n \log n}$

Thm: [Hopkins–Raghavendra–Shetty'22] conjecture holds if $\mathrm{rank}(\mathbf{A}_i) \ll \sqrt{n}$



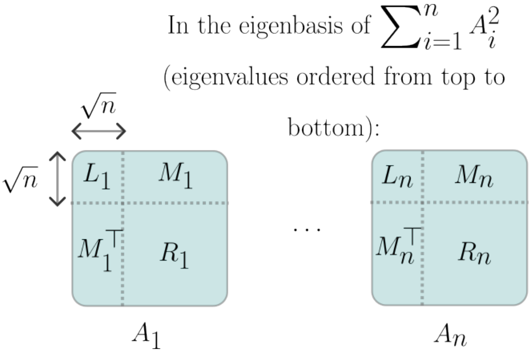
1. Apply $\ell_{\frac{1}{2}}$-regularization to $\mathbf{L}_1, \ldots, \mathbf{L}_n$ and $\mathbf{M}_1\mathbf{M}_1^\top, \ldots, \mathbf{M}_n\mathbf{M}_n^\top$.

Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{R}^{n\times n}$ s.t. $\|\mathbf{A}_i\|_{\mathrm{op}} \leqslant 1$, there exists $x_1, \ldots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{\mathrm{op}} = O(\sqrt{n})$

Random coloring: $\sqrt{\log n} \cdot \|\sum_i \mathbf{A}_i^2\|_{\mathrm{op}}^{\frac{1}{2}} \leqslant \sqrt{n \log n}$

Thm: [Hopkins–Raghavendra–Shetty'22] conjecture holds if $\mathrm{rank}(\mathbf{A}_i) \ll \sqrt{n}$



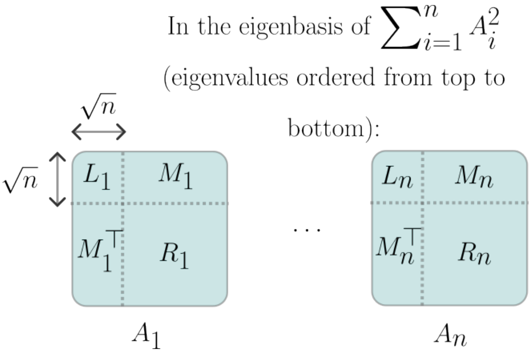
1. Apply $\ell_{\frac{1}{2}}$-regularization to $\mathbf{L}_1, \ldots, \mathbf{L}_n$ and $\mathbf{M}_1\mathbf{M}_1^\top, \ldots, \mathbf{M}_n\mathbf{M}_n^\top$.
2. $\|\sum_i \mathbf{R}_i^2\|_{\mathrm{op}} \leqslant \frac{n}{\log n}$: apply derandomized matrix Chernoff to $\mathbf{R}_1, \ldots, \mathbf{R}_n$.

Matrix Spencer conjecture

Conjecture: For any symmetric $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathbb{R}^{n \times n}$ s.t. $\|\mathbf{A}_i\|_{op} \leqslant 1$, there exists $x_1, \dots, x_n \in \{-1, 1\}$ s.t. $\|\sum_i x_i \mathbf{A}_i\|_{op} = O(\sqrt{n})$

Random coloring: $\sqrt{\log n} \cdot \|\sum_i \mathbf{A}_i^2\|_{op}^{\frac{1}{2}} \leqslant \sqrt{n \log n}$

Thm: [Hopkins–Raghavendra–Shetty'22] conjecture holds if $\mathrm{rank}(\mathbf{A}_i) \ll \sqrt{n}$



1. Apply $\ell_{\frac{1}{2}}$-regularization to $\mathbf{L}_1, \dots, \mathbf{L}_n$ and $\mathbf{M}_1\mathbf{M}_1^\top, \dots, \mathbf{M}_n\mathbf{M}_n^\top$.
2. $\|\sum_i \mathbf{R}_i^2\|_{op} \leqslant \frac{n}{\log n}$: apply derandomized matrix Chernoff to $\mathbf{R}_1, \dots, \mathbf{R}_n$.

Open question 3: same if $\mathrm{rank}(\mathbf{A}_i) \ll n$? [Bansal–Jiang–Meka'23]

Optimizing a random quadratic form

Problem: solve for $\mathbf{x}$ in

$$\max_{\mathbf{x}\in\{-1,1\}^n} \frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \qquad ,$$

where $\mathbf{A}$ is a random symmetric matrix with i.i.d. $\pm 1/\sqrt{n}$ entries

Optimizing a random quadratic form

Problem: solve for $\mathbf{x}$ in

$$\max_{\mathbf{x}\in\{-1,1\}^n} \frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \approx P_* \text{ for some } P_* \approx \ldots,$$

where $\mathbf{A}$ is a random symmetric matrix with i.i.d. $\pm 1/\sqrt{n}$ entries

Optimizing a random quadratic form

Problem: solve for $\mathbf{x}$ in

$$\max_{\mathbf{x}\in\{-1,1\}^n} \frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \approx P_* \text{ for some } P_* \approx \ldots,$$

where $\mathbf{A}$ is a random symmetric matrix with i.i.d. $\pm 1/\sqrt{n}$ entries

[Montanari'19; El Alaoui–Montanari–Sellke'21] algorithm achieving $P_* - \varepsilon$ whp

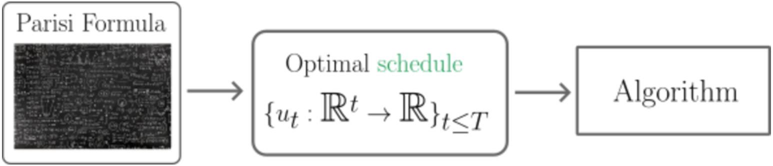
Optimizing a random quadratic form

Problem: solve for $\mathbf{x}$ in

$$\max_{\mathbf{x}\in\{-1,1\}^n} \frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \approx P_* \text{ for some } P_* \approx \ldots,$$

where $\mathbf{A}$ is a random symmetric matrix with i.i.d. $\pm 1/\sqrt{n}$ entries

[Montanari'19; El Alaoui–Montanari–Sellke'21] algorithm achieving $P_* - \varepsilon$ whp



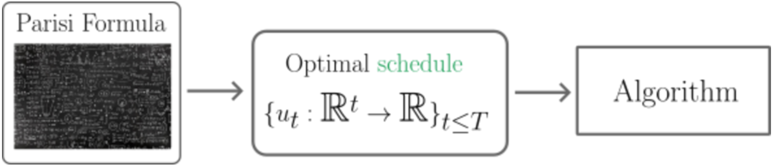
Optimizing a random quadratic form

Problem: solve for **x** in

$$\max_{\mathbf{x}\in\{-1,1\}^n} \frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \approx P_* \text{ for some } P_* \approx \ldots,$$

where **A** is a random symmetric matrix with i.i.d. $\pm 1/\sqrt{n}$ entries

[Montanari'19; El Alaoui–Montanari–Sellke'21] algorithm achieving $P_* - \varepsilon$ whp



Optimal schedule: given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t\leqslant T} 2\mathbb{E}[u_t(g_0,\ldots,g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t\leqslant T} u_t(g_0,\ldots,g_{t-1})g_t \in [-1,1] \text{ a.s.}$$

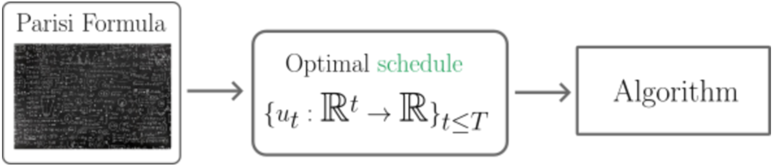
Optimizing a random quadratic form

Problem: solve for $\mathbf{x}$ in

$$\max_{\mathbf{x}\in\{-1,1\}^n} \frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \approx P_* \text{ for some } P_* \approx \ldots,$$

where $\mathbf{A}$ is a random symmetric matrix with i.i.d. $\pm 1/\sqrt{n}$ entries

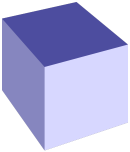
[Montanari'19; El Alaoui–Montanari–Sellke'21] algorithm achieving $P_* - \varepsilon$ whp



Optimal schedule: given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t\leqslant T} 2\mathbb{E}[u_t(g_0,\ldots,g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t\leqslant T} u_t(g_0,\ldots,g_{t-1})g_t \in [-1,1] \text{ a.s.}$$



Question: How to pick the g_t's?

Diagram analysis of iterative algorithms

Perspective: “symmetrized” Fourier analysis

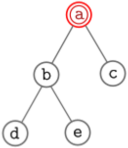
Diagram analysis of iterative algorithms

Perspective: "symmetrized" Fourier analysis

$$Z_a^{\alpha} = \sum_{\substack{b:\\ a,b \text{ distinct}}} A_{ab}$$

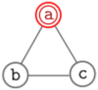
α

$$Z_a^{\beta} = \sum_{\substack{b,c,d,e:\\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



β

$$Z_a^{\gamma} = \sum_{\substack{b,c:\\ a,b,c \text{ distinct}}} A_{ab}A_{ac}A_{bc}$$



γ

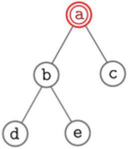
Diagram analysis of iterative algorithms

Perspective: "symmetrized" Fourier analysis

$$Z_a^\alpha = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

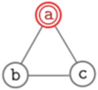
α

$$Z_a^\beta = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab} A_{bd} A_{be} A_{ac}$$



β

$$Z_a^\gamma = \sum_{\substack{b,c: \\ a,b,c \text{ distinct}}} A_{ab} A_{ac} A_{bc}$$



γ

Facts: if **A** is symmetric with i.i.d. $\pm 1/\sqrt{n}$ entries,

1. The diagrams are orthogonal w.r.t. $\mathbb{E}_A$.

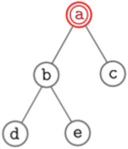
Diagram analysis of iterative algorithms

Perspective: "symmetrized" Fourier analysis

$$Z_a^\alpha = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

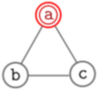
α

$$Z_a^\beta = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



β

$$Z_a^\gamma = \sum_{\substack{b,c: \\ a,b,c \text{ distinct}}} A_{ab}A_{ac}A_{bc}$$



γ

Facts: if **A** is symmetric with i.i.d. $\pm 1/\sqrt{n}$ entries,

1. The diagrams are orthogonal w.r.t. $\mathbb{E}_A$.
2. As $n \to \infty$, the only non-negligible diagrams are the trees.

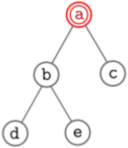
Diagram analysis of iterative algorithms

Perspective: "symmetrized" Fourier analysis

$$Z_a^{\alpha} = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

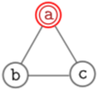
α

$$Z_a^{\beta} = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



β

$$Z_a^{\gamma} = \sum_{\substack{b,c: \\ a,b,c \text{ distinct}}} A_{ab}A_{ac}A_{bc}$$



γ

Facts: if $\mathbf{A}$ is symmetric with i.i.d. $\pm 1/\sqrt{n}$ entries,

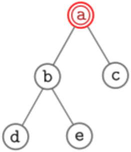
1. The diagrams are orthogonal w.r.t. $\mathbb{E}_A$.
2. As $n \to \infty$, the only non-negligible diagrams are the trees.
3. As $n \to \infty$, the trees where the root has degree 1 are independent Gaussian vectors.

Asymptotic diagrams cheatsheet (1/2)

$$Z_a^\alpha = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

α

$$Z_a^\beta = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



β

Asymptotic diagram basis $\{\mathbf{Z}^\alpha : \alpha \text{ rooted tree}\}$

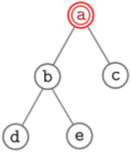
Rule 1: multiply diagrams coordinatewise

Asymptotic diagrams cheatsheet (1/2)

$$Z_a^\alpha = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

α

$$Z_a^\beta = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$

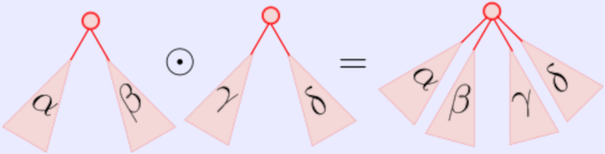


β

Asymptotic diagram basis $\{\mathbf{Z}^\alpha : \alpha \text{ rooted tree}\}$

Rule 1: multiply diagrams coordinatewise

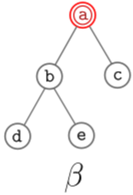
$$\text{if } \{\alpha, \beta\} \cap \{\gamma, \delta\} = \emptyset:$$



Asymptotic diagrams cheatsheet (1/2)

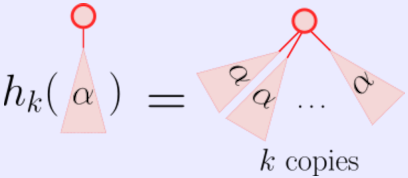
$$Z_a^\alpha = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

$$Z_a^\beta = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



Asymptotic diagram basis $\{\mathbf{Z}^\alpha : \alpha \text{ rooted tree}\}$

Ex:

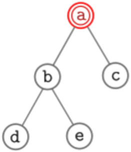


Asymptotic diagrams cheatsheet (2/2)

$$Z_a^{\alpha} = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

α

$$Z_a^{\beta} = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



β

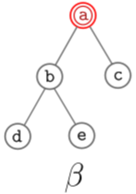
Asymptotic diagram basis $\{\mathbf{Z}^{\alpha} : \alpha$ rooted tree$\}$

Rule 2: multiply by **A**

Asymptotic diagrams cheatsheet (2/2)

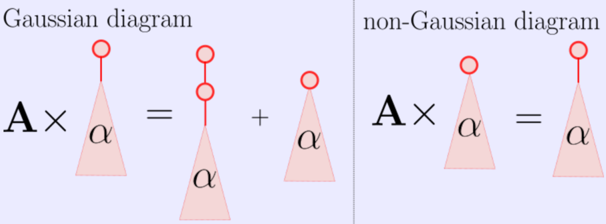
$$Z_a^{\alpha} = \sum_{\substack{b: \\ a,b \text{ distinct}}} A_{ab}$$

$$Z_a^{\beta} = \sum_{\substack{b,c,d,e: \\ a,b,c,d,e \text{ distinct}}} A_{ab}A_{bd}A_{be}A_{ac}$$



Asymptotic diagram basis $\{\mathbf{Z}^{\alpha} : \alpha \text{ rooted tree}\}$

Rule 2: multiply by **A**



Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \ldots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \ldots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \ldots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \ldots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \ldots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \ldots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

- *Rule 1*: $u_t(\mathbf{g}_0, \ldots, \mathbf{g}_{t-1})$ has full mass on diagrams of depth $\leqslant t-1$.

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \dots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \dots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

- *Rule 1*: $u_t(\mathbf{g}_0, \dots, \mathbf{g}_{t-1})$ has full mass on diagrams of depth $\leqslant t-1$.
- *Rule 1*: $\mathbf{u}_t \odot \mathbf{g}_t$ has full mass on depth-t diagrams.

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \ldots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \ldots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

- *Rule 1*: $u_t(\mathbf{g}_0, \ldots, \mathbf{g}_{t-1})$ has full mass on diagrams of depth $\leqslant t-1$.
- *Rule 1*: $\mathbf{u}_t \odot \mathbf{g}_t$ has full mass on depth-t diagrams.
- *Rule 2*: $\mathbf{A}(\mathbf{u}_t \odot \mathbf{g}_t)$ has full mass on Gaussian diagrams of depth $t+1$ and diagrams of depth $t-1$. Define $\mathbf{g}_{t+1}$ to be the depth-$(t+1)$ part.

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \dots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \dots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

- *Rule 1*: $u_t(\mathbf{g}_0, \dots, \mathbf{g}_{t-1})$ has full mass on diagrams of depth $\leqslant t-1$.
- *Rule 1*: $\mathbf{u}_t \odot \mathbf{g}_t$ has full mass on depth-t diagrams.
- *Rule 2*: $\mathbf{A}(\mathbf{u}_t \odot \mathbf{g}_t)$ has full mass on Gaussian diagrams of depth $t+1$ and diagrams of depth $t-1$. Define $\mathbf{g}_{t+1}$ to be the depth-$(t+1)$ part.

If $\mathbf{x} = \sum_{t \leqslant T} \mathbf{u}_t \odot \mathbf{g}_t$,

$$\mathbb{E}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle = \sum_{s,t \leqslant T} \mathbb{E} \langle \mathbf{u}_s \odot \mathbf{g}_s, \mathbf{A}\mathbf{u}_t \odot \mathbf{g}_t \rangle$$

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \ldots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \ldots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

- *Rule 1*: $u_t(\mathbf{g}_0, \ldots, \mathbf{g}_{t-1})$ has full mass on diagrams of depth $\leqslant t-1$.
- *Rule 1*: $\mathbf{u}_t \odot \mathbf{g}_t$ has full mass on depth-t diagrams.
- *Rule 2*: $\mathbf{A}(\mathbf{u}_t \odot \mathbf{g}_t)$ has full mass on Gaussian diagrams of depth $t+1$ and diagrams of depth $t-1$. Define $\mathbf{g}_{t+1}$ to be the depth-$(t+1)$ part.

If $\mathbf{x} = \sum_{t \leqslant T} \mathbf{u}_t \odot \mathbf{g}_t$,

$$\mathbb{E}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle = \sum_{s,t \leqslant T} \mathbb{E}\left\langle \mathbf{u}_s \odot \mathbf{g}_s, \mathbf{A}\mathbf{u}_t \odot \mathbf{g}_t \right\rangle = 2 \sum_{t \leqslant T} \mathbb{E}\langle \mathbf{u}_t \odot \mathbf{g}_t, \mathbf{g}_t\rangle$$

Back to optimizing random quadratic forms

Input: optimal schedule $\{u_t : \mathbb{R}^t \to \mathbb{R}\}_{t \leqslant T}$ s.t. given i.i.d. $g_t \sim \mathcal{N}(0,1)$,

$$\sum_{t \leqslant T} 2\mathbb{E}[u_t(g_0, \dots, g_{t-1})] \geqslant P_* - \varepsilon$$

$$\sum_{t \leqslant T} u_t(g_0, \dots, g_{t-1}) g_t \in [-1, 1] \text{ a.s.}$$

Output: $\mathbf{x} \in [-1,1]^n$ s.t. $\frac{1}{n}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geqslant P_* - \varepsilon$.

Pick $\mathbf{g}_t$ among depth-t Gaussian diagrams!

- *Rule 1*: $u_t(\mathbf{g}_0, \dots, \mathbf{g}_{t-1})$ has full mass on diagrams of depth $\leqslant t-1$.
- *Rule 1*: $\mathbf{u}_t \odot \mathbf{g}_t$ has full mass on depth-t diagrams.
- *Rule 2*: $\mathbf{A}(\mathbf{u}_t \odot \mathbf{g}_t)$ has full mass on Gaussian diagrams of depth $t+1$ and diagrams of depth $t-1$. Define $\mathbf{g}_{t+1}$ to be the depth-$(t+1)$ part.

If $\mathbf{x} = \sum_{t \leqslant T} \mathbf{u}_t \odot \mathbf{g}_t$,

$$\mathbb{E}\langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle = \sum_{s,t \leqslant T} \mathbb{E}\left\langle \mathbf{u}_s \odot \mathbf{g}_s, \mathbf{A}\mathbf{u}_t \odot \mathbf{g}_t \right\rangle = 2 \sum_{t \leqslant T} \mathbb{E}\langle \mathbf{u}_t \odot \mathbf{g}_t, \mathbf{g}_t \rangle = n \sum_{t \leqslant T} 2\mathbb{E}[u_t]. \quad \square$$

Upcoming challenges

Applies to *generalized first order methods* [Montanari–Celentano–Wu'20]

Ex: approximate message passing, power iteration, ...

Upcoming challenges

Applies to *generalized first order methods* [Montanari–Celentano–Wu'20]

Ex: approximate message passing, power iteration, ...

Theorem 3.35 (State evolution for GFOM). *Assume Assumption 3.1 on A and Assumption 3.34 on $f_0, f_1, \ldots, f_t$. Generate $x_t, y_t \in \mathbb{R}^n$ using the GFOM*

$$x_0 = \vec{1} \qquad y_t = Ax_t \qquad x_{t+1} = f_t(y_t, \ldots, y_0)\,.$$

Let $X_t, Y_t \in \Omega$ be the result of running the GFOM operations asymptotically using the rules in Section 2.3. Then for all polynomial functions $\psi : \mathbb{R}^{2(t+1)} \to \mathbb{R}$,

$$\frac{1}{n}\sum_{i=1}^{n} \psi(x_{t,i}, y_{t,i}, \ldots, x_{0,i}, y_{0,i}) \xrightarrow{a.s.} \mathbb{E}[\psi(X_t, Y_t, \ldots, X_0, Y_0)]\,.$$

Furthermore, X_t, Y_t are universal (they do not depend on the distributions μ or μ_0 in Assumption 3.1).

Upcoming challenges

Applies to *generalized first order methods* [Montanari–Celentano–Wu'20]

Ex: approximate message passing, power iteration, ...

Theorem 3.35 (State evolution for GFOM). *Assume Assumption 3.1 on A and Assumption 3.34 on $f_0, f_1, \ldots, f_t$. Generate $x_t, y_t \in \mathbb{R}^n$ using the GFOM*

$$x_0 = \vec{1} \qquad y_t = Ax_t \qquad x_{t+1} = f_t(y_t, \ldots, y_0)\,.$$

Let $X_t, Y_t \in \Omega$ be the result of running the GFOM operations asymptotically using the rules in Section 2.3. Then for all polynomial functions $\psi : \mathbb{R}^{2(t+1)} \to \mathbb{R}$,

$$\frac{1}{n}\sum_{i=1}^{n} \psi(x_{t,i}, y_{t,i}, \ldots, x_{0,i}, y_{0,i}) \xrightarrow{a.s.} \mathbb{E}[\psi(X_t, Y_t, \ldots, X_0, Y_0)]\,.$$

Furthermore, X_t, Y_t are universal (they do not depend on the distributions μ or μ_0 in Assumption 3.1).

Open question 4: how about $\Omega(\log n)$ iterations?

Conclusion

Summary:

- Discrepancy problems and their applications to sparsification
- Newton's method on a regularized objective for Spencer's theorem
- Combining regularizers for the low-rank matrix Spencer conjecture
- Optimizing a random quadratic form and the "optimal schedules" from Parisi formula
- Diagram analysis of iterative algorithms on random matrices

Conclusion

Summary:

- Discrepancy problems and their applications to sparsification
- Newton's method on a regularized objective for Spencer's theorem
- Combining regularizers for the low-rank matrix Spencer conjecture
- Optimizing a random quadratic form and the "optimal schedules" from Parisi formula
- Diagram analysis of iterative algorithms on random matrices

Open question 1: tight constant in Spencer's theorem?

Open question 2: other useful regularizers?

Open question 3: matrix Spencer up to polylog rank using regularized Newton's method?

Open question 4: diagram analysis of iterative algorithms for $\Omega(\log n)$ iterations?