New algorithms for approximating massive datasets

Naren Sarayu Manoj (https://nsmanoj.com)

TTIC

2024 Feb 21

This talk – geometrically summarizing massive datasets

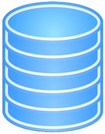
Guiding questions/motivations

Can we:

Guiding questions/motivations

Can we:

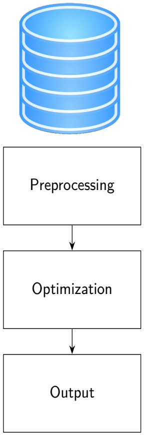
- quickly summarize a massive dataset in a geometrically meaningful way?



Guiding questions/motivations

Can we:

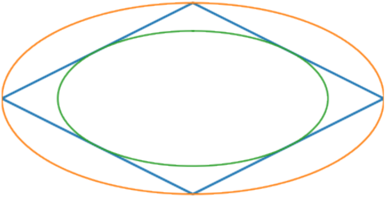
- quickly summarize a massive dataset in a geometrically meaningful way?
- use our summaries as optimization primitives or preprocessing routines?

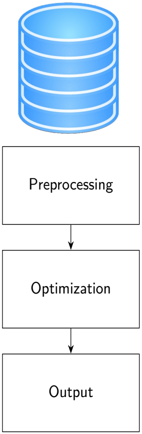


Guiding questions/motivations

Can we:

- quickly summarize a massive dataset in a geometrically meaningful way?
- use our summaries as optimization primitives or preprocessing routines?
- gain a better understanding of high-dimensional convex geometry?





Streaming algorithms for ellipsoidal approximation of convex polytopes – symmetric and asymmetric

https://arxiv.org/abs/2206.07250 (COLT 2022) and
https://arxiv.org/abs/2311.09460 (STOC 2024, to appear)

Yury Makarychev, *NSM*, Max Ovsiankin

Table of contents

Ellipsoidal approximations

Basic problem

Given a convex body $X \subset \mathbb{R}^n$, find an ellipsoid $\mathcal{E}$ and a center $\boldsymbol{c} \in \mathbb{R}^n$ such that $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$.

Basic problem

Given a convex body $X \subset \mathbb{R}^n$, find an ellipsoid $\mathcal{E}$ and a center $\boldsymbol{c} \in \mathbb{R}^n$ such that $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$.

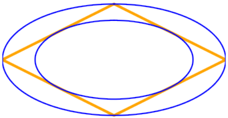


Figure: $\mathcal{E}/\sqrt{2} \subseteq X \subseteq \mathcal{E}$

Ellipsoidal approximations

Basic problem

Given a convex body $X \subset \mathbb{R}^n$, find an ellipsoid $\mathcal{E}$ and a center $\boldsymbol{c} \in \mathbb{R}^n$ such that $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$.

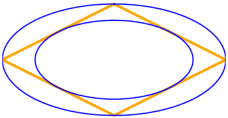


Figure: $\mathcal{E}/\sqrt{2} \subseteq X \subseteq \mathcal{E}$

Such an $\mathcal{E}$ allows us to succinctly represent a convex body (n^2 floats).

Offline solution

Ellipsoidal approximation – offline

Given a convex body $X \subseteq \mathbb{R}^n$, compute a center $\boldsymbol{c} \in \mathbb{R}^n$ and an ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α. Let α be $\mathcal{E}$'s *approximation factor*.

Offline solution

Ellipsoidal approximation – offline

Given a convex body $X \subseteq \mathbb{R}^n$, compute a center $\boldsymbol{c} \in \mathbb{R}^n$ and an ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α. Let α be $\mathcal{E}$'s *approximation factor*.

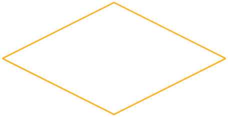
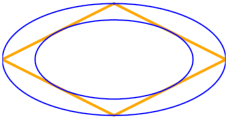


Figure: X

Offline solution

Ellipsoidal approximation – offline

Given a convex body $X \subseteq \mathbb{R}^n$, compute a center $\boldsymbol{c} \in \mathbb{R}^n$ and an ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α. Let α be $\mathcal{E}$'s *approximation factor*.

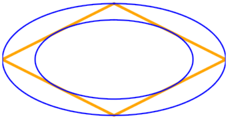


$${}^{\mathcal{E}}/_{\sqrt{2}} \subseteq X \subseteq \mathcal{E}$$

Offline solution

Ellipsoidal approximation – offline

Given a convex body $X \subseteq \mathbb{R}^n$, compute a center $\boldsymbol{c} \in \mathbb{R}^n$ and an ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α. Let α be $\mathcal{E}$'s *approximation factor*.



$$\mathcal{E}/\sqrt{2} \subseteq X \subseteq \mathcal{E}$$

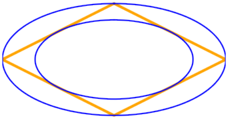
Theorem (John [Joh48])

If $\mathcal{E}$ is the minimum volume ellipsoid covering X, then we can always achieve $\alpha \leq n$.

Offline solution

Ellipsoidal approximation – offline

Given a convex body $X \subseteq \mathbb{R}^n$, compute a center $\boldsymbol{c} \in \mathbb{R}^n$ and an ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α. Let α be $\mathcal{E}$'s *approximation factor*.



$$\mathcal{E}/\sqrt{2} \subseteq X \subseteq \mathcal{E}$$

Theorem (John [Joh48])

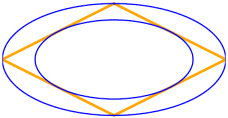
If $\mathcal{E}$ is the minimum volume ellipsoid covering X, then we can always achieve $\alpha \leq n$. If X is origin-symmetric, then we can always achieve $\alpha \leq \sqrt{n}$.

There exists X for which *any* ellipsoidal approximation for X must achieve $\alpha = n$ (e.g. simplex).

Offline solution

Ellipsoidal approximation – offline

Given a convex body $X \subseteq \mathbb{R}^n$, compute a center $\boldsymbol{c} \in \mathbb{R}^n$ and an ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α. Let α be $\mathcal{E}$'s *approximation factor*.



$$\mathcal{E}/\sqrt{2} \subseteq X \subseteq \mathcal{E}$$

Theorem (John [Joh48])

If $\mathcal{E}$ is the minimum volume ellipsoid covering X, then we can always achieve $\alpha \leq n$. If X is origin-symmetric, then we can always achieve $\alpha \leq \sqrt{n}$.

There exists X for which *any* ellipsoidal approximation for X must achieve $\alpha = n$ (e.g. simplex). For symmetric polyhedrons with m linear constraints, John's Ellipsoid can be approximated in time $\widetilde{O}\left(mn^2\right)$ [CCLY19].

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \text{conv}\,(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}\!/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$\boxed{\boldsymbol{x}_1}$

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + \mathcal{E}/\alpha \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$$\boxed{\boldsymbol{x}_1} \to \boxed{\boldsymbol{x}_2}$$

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$$\boxed{\boldsymbol{x}_1} \to \boxed{\boldsymbol{x}_2} \to \cdots \to \boxed{\boldsymbol{x}_m}$$

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$$\boxed{\boldsymbol{x}_1} \to \boxed{\boldsymbol{x}_2} \to \cdots \to \boxed{\boldsymbol{x}_m}$$

Motivation – Suppose we want to summarize a dataset in a resource-constrained environment.

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$$\boxed{\boldsymbol{x}_1} \to \boxed{\boldsymbol{x}_2} \to \cdots \to \boxed{\boldsymbol{x}_m}$$

Motivation – Suppose we want to summarize a dataset in a resource-constrained environment.

We want our algorithms to be efficient:

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}\!/\!_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$$\boxed{\boldsymbol{x}_1} \to \boxed{\boldsymbol{x}_2} \to \cdots \to \boxed{\boldsymbol{x}_m}$$

Motivation – Suppose we want to summarize a dataset in a resource-constrained environment.

We want our algorithms to be efficient:

- Cannot store too many points at once;

Streaming/online ellipsoidal approximations

Problem

Given a convex body $X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ as a stream of points, find a center $\boldsymbol{c} \in \mathbb{R}^n$ and ellipsoid $\mathcal{E}$ with $\boldsymbol{c} + {}^{\mathcal{E}}/_{\alpha} \subseteq X \subseteq \boldsymbol{c} + \mathcal{E}$ that minimizes α.

$$\boxed{\boldsymbol{x}_1} \to \boxed{\boldsymbol{x}_2} \to \cdots \to \boxed{\boldsymbol{x}_m}$$

Motivation – Suppose we want to summarize a dataset in a resource-constrained environment.

We want our algorithms to be efficient:

- Cannot store too many points at once;
- Update time in each iteration must be fast.

Table of contents

Aspect ratio of convex body

Aspect ratio of convex body

Let R be the smallest value and r be the largest value such that:

$$r \cdot B_2^n \subseteq X \subseteq R \cdot B_2^n$$

The *aspect ratio* of X is $\kappa(X) := {}^{R}/_{r}$.

Aspect ratio of convex body

Aspect ratio of convex body

Let R be the smallest value and r be the largest value such that:

$$r \cdot B_2^n \subseteq X \subseteq R \cdot B_2^n$$

The *aspect ratio* of X is $\kappa(X) := {}^{R}/_{r}$.

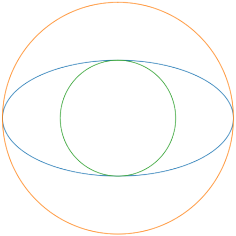


Figure: $1 \cdot B_2^n \subseteq \mathcal{E} \subseteq 2 \cdot B_2^n \Rightarrow \kappa(\mathcal{E}) = {}^{2}/_{1}$

Aspect ratio of convex body

Let R be the smallest value and r be the largest value such that:

$$r \cdot B_2^n \subseteq X \subseteq R \cdot B_2^n$$

The *aspect ratio* of X is $\kappa(X) := {}^{R}\!/_{r}$.

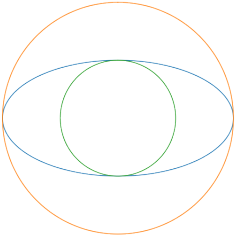


Figure: $1 \cdot B_2^n \subseteq \mathcal{E} \subseteq 2 \cdot B_2^n \Rightarrow \kappa(\mathcal{E}) = {}^{2}\!/_{1}$

Example

- For an ellipsoid $\mathcal{E}$ with major axis $\lambda_{\max}$ and minor axis $\lambda_{\min}$, we have $\kappa(\mathcal{E}) = {}^{\lambda_{\max}}\!/_{\lambda_{\min}}$;

Aspect ratio of convex body

Aspect ratio of convex body

Let R be the smallest value and r be the largest value such that:

$$r \cdot B_2^n \subseteq X \subseteq R \cdot B_2^n$$

The *aspect ratio* of X is $\kappa(X) := {}^R/_r$.

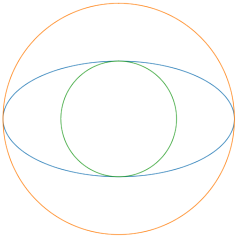


Figure: $1 \cdot B_2^n \subseteq \mathcal{E} \subseteq 2 \cdot B_2^n \Rightarrow \kappa(\mathcal{E}) = {}^2/_1$

Example

- For an ellipsoid $\mathcal{E}$ with major axis $\lambda_{\max}$ and minor axis $\lambda_{\min}$, we have $\kappa(\mathcal{E}) = {}^{\lambda_{\max}}/_{\lambda_{\min}}$;
- $\kappa(B_1^n) = \sqrt{n}$;

Aspect ratio of convex body

Let R be the smallest value and r be the largest value such that:

$$r \cdot B_2^n \subseteq X \subseteq R \cdot B_2^n$$

The *aspect ratio* of X is $\kappa(X) := {}^{R}/_{r}$.

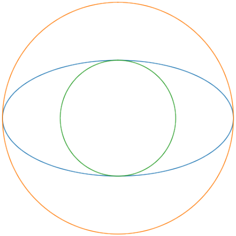


Figure: $1 \cdot B_2^n \subseteq \mathcal{E} \subseteq 2 \cdot B_2^n \Rightarrow \kappa(\mathcal{E}) = {}^{2}/_{1}$

Example

- For an ellipsoid $\mathcal{E}$ with major axis $\lambda_{\max}$ and minor axis $\lambda_{\min}$, we have $\kappa(\mathcal{E}) = {}^{\lambda_{\max}}/_{\lambda_{\min}}$;
- $\kappa(B_1^n) = \sqrt{n}$;
- $\kappa(B_2^n) = 1$

Results for streaming ellipsoidal approximations

Approximation result

Results for streaming ellipsoidal approximations

Approximation result

Assumption: given $\boldsymbol{c}_0 \in \mathbb{R}^n$ and $r_0 > 0$ such that $\boldsymbol{c}_0 + r_0 \cdot B_2^n \subseteq X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$.

Results for streaming ellipsoidal approximations

Approximation result

Assumption: given $\boldsymbol{c}_0 \in \mathbb{R}^n$ and $r_0 > 0$ such that $\boldsymbol{c}_0 + r_0 \cdot B_2^n \subseteq X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$.

Let $R(X)$ be such that there exists $\boldsymbol{c}_R$ such that $X \subseteq \boldsymbol{c}_R + R(X) \cdot B_2^n$.

Approximation result

Assumption: given $\boldsymbol{c}_0 \in \mathbb{R}^n$ and $r_0 > 0$ such that $\boldsymbol{c}_0 + r_0 \cdot B_2^n \subseteq X = \operatorname{conv}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_m)$.

Let $R(X)$ be such that there exists $\boldsymbol{c}_R$ such that $X \subseteq \boldsymbol{c}_R + R(X) \cdot B_2^n$.

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

Results for streaming ellipsoidal approximations

Approximation result

Assumption: given $\boldsymbol{c}_0 \in \mathbb{R}^n$ and $r_0 > 0$ such that $\boldsymbol{c}_0 + r_0 \cdot B_2^n \subseteq X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$.

Let $R(X)$ be such that there exists $\boldsymbol{c}_R$ such that $X \subseteq \boldsymbol{c}_R + R(X) \cdot B_2^n$.

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

$$\boldsymbol{c}_m + \frac{\mathcal{E}_m}{Cn\log(R(X)/r_0 + 1)} \subseteq X \subseteq \boldsymbol{c}_m + \mathcal{E}_m \qquad \text{(asymmetric)}$$

$$\frac{\mathcal{E}_m}{C\sqrt{n\log(R(X)/r_0 + 1)}} \subseteq X \subseteq \mathcal{E}_m \qquad \text{(symmetric)}$$

Results for streaming ellipsoidal approximations

Approximation result

Assumption: given $\boldsymbol{c}_0 \in \mathbb{R}^n$ and $r_0 > 0$ such that $\boldsymbol{c}_0 + r_0 \cdot B_2^n \subseteq X = \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$.

Let $R(X)$ be such that there exists $\boldsymbol{c}_R$ such that $X \subseteq \boldsymbol{c}_R + R(X) \cdot B_2^n$.

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

$$\boldsymbol{c}_m + \frac{\mathcal{E}_m}{Cn\log(R(X)/r_0 + 1)} \subseteq X \subseteq \boldsymbol{c}_m + \mathcal{E}_m \qquad \text{(asymmetric)}$$

$$\frac{\mathcal{E}_m}{C\sqrt{n\log(R(X)/r_0 + 1)}} \subseteq X \subseteq \mathcal{E}_m \qquad \text{(symmetric)}$$

Runtime: $\widetilde{O}(mn^2)$. **Space complexity**: $O(n^2)$ floating point numbers.

Results for streaming ellipsoidal approximations

Approximation result

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

$$\boldsymbol{c}_m + \frac{\mathcal{E}_m}{Cn\log(R(X)/r_0+1)} \subseteq X \subseteq \boldsymbol{c}_m + \mathcal{E}_m \qquad \text{(asymmetric)}$$

$$\frac{\mathcal{E}_m}{C\sqrt{n\log(R(X)/r_0+1)}} \subseteq X \subseteq \mathcal{E}_m \qquad \text{(symmetric)}$$

Runtime: $\widetilde{O}(mn^2)$. **Space complexity**: $O(n^2)$ floating point numbers.

Results for streaming ellipsoidal approximations

Approximation result

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

$$\boldsymbol{c}_m + \frac{\mathcal{E}_m}{Cn\log(R(X)/r_0+1)} \subseteq X \subseteq \boldsymbol{c}_m + \mathcal{E}_m \qquad \text{(asymmetric)}$$

$$\frac{\mathcal{E}_m}{C\sqrt{n\log(R(X)/r_0+1)}} \subseteq X \subseteq \mathcal{E}_m \qquad \text{(symmetric)}$$

Runtime: $\widetilde{O}(mn^2)$. **Space complexity**: $O(n^2)$ floating point numbers.

- By John's Theorem, the dependence on dimension n is optimal.

Results for streaming ellipsoidal approximations

Approximation result

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

$$\boldsymbol{c}_m + \frac{\mathcal{E}_m}{Cn\log(R(X)/r_0 + 1)} \subseteq X \subseteq \boldsymbol{c}_m + \mathcal{E}_m \qquad \text{(asymmetric)}$$

$$\frac{\mathcal{E}_m}{C\sqrt{n\log(R(X)/r_0 + 1)}} \subseteq X \subseteq \mathcal{E}_m \qquad \text{(symmetric)}$$

Runtime: $\widetilde{O}(mn^2)$. **Space complexity**: $O(n^2)$ floating point numbers.

- By John's Theorem, the dependence on dimension n is optimal.
- If X is asymmetric, then for all "monotone" algorithms, there exists an example stream for which

$$\alpha = \Omega\left(\frac{n\log(R(X)/r_0)}{\log n}\right)$$

Results for streaming ellipsoidal approximations

Approximation result

There exist algorithms that output a center $\boldsymbol{c}_m$ and ellipsoid $\mathcal{E}_m \supseteq X$ such that:

$$\boldsymbol{c}_m + \frac{\mathcal{E}_m}{Cn\log(R(X)/r_0+1)} \subseteq X \subseteq \boldsymbol{c}_m + \mathcal{E}_m \qquad \text{(asymmetric)}$$

$$\frac{\mathcal{E}_m}{C\sqrt{n\log(R(X)/r_0+1)}} \subseteq X \subseteq \mathcal{E}_m \qquad \text{(symmetric)}$$

Runtime: $\widetilde{O}(mn^2)$. **Space complexity**: $O(n^2)$ floating point numbers.

- By John's Theorem, the dependence on dimension n is optimal.
- If X is asymmetric, then for all "monotone" algorithms, there exists an example stream for which

$$\alpha = \Omega\left(\frac{n\log(R(X)/r_0)}{\log n}\right)$$

- If we assume the points $\boldsymbol{x}_i$ have integer coordinates in $[-N,\dots,N]$, then we can replace $\log(R/r)$ with $\log n + \log N$.

Table of contents

Greedy algorithm, symmetric case

Assumption

The algorithm is told a value of r_0 *such that* $r_0 \cdot B_2^n \subseteq X$.

Greedy algorithm, symmetric case

Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Greedy algorithm, symmetric case

Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

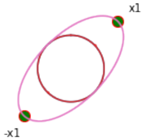
Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.

Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.

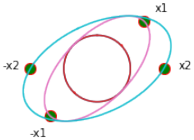


Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.

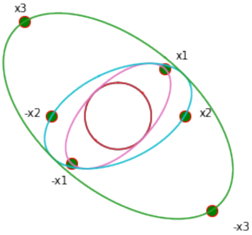


Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.

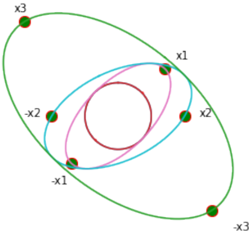


Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.



Formally, if $\mathcal{E}_t = \left\{ \boldsymbol{x} \; : \; \|\mathbf{A}_t \boldsymbol{x}\|_2 \leq 1 \right\}$ for invertible $\mathbf{A}_t$, then:

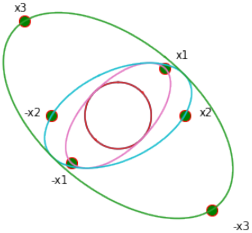
Greedy algorithm, symmetric case

Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.



Formally, if $\mathcal{E}_t = \left\{ \boldsymbol{x} \; : \; \|\mathbf{A}_t \boldsymbol{x}\|_2 \leq 1 \right\}$ for invertible $\mathbf{A}_t$, then:

$$\mathbf{A}_t = \mathbf{A}_{t-1} - \left(1 - \frac{1}{\|\mathbf{A}_{t-1}\boldsymbol{x}_t\|_2}\right) \left(\frac{\mathbf{A}_{t-1}\boldsymbol{x}_t}{\|\mathbf{A}_{t-1}\boldsymbol{x}_t\|_2}\right) \left(\frac{\mathbf{A}_{t-1}\boldsymbol{x}_t}{\|\mathbf{A}_{t-1}\boldsymbol{x}_t\|_2}\right)^T \mathbf{A}_{t-1}$$

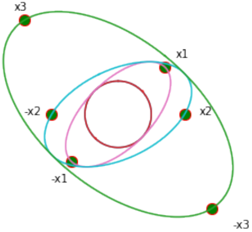
Greedy algorithm, symmetric case

Assumption

The algorithm is told a value of r_0 such that $r_0 \cdot B_2^n \subseteq X$.

Initialization – Let $\mathcal{E}_0 = r \cdot B_2^n$.

Update rule – given ellipsoid $\mathcal{E}_{t-1}$ and new point $\pm \boldsymbol{x}_t$, let $\mathcal{E}_t$ be the minimum volume origin-centered ellipsoid covering $\mathcal{E}_{t-1}$ and $\pm \boldsymbol{x}_t$.



Formally, if $\mathcal{E}_t = \left\{\boldsymbol{x} \;:\; \|\mathbf{A}_t \boldsymbol{x}\|_2 \leq 1\right\}$ for invertible $\mathbf{A}_t$, then:

$$\mathbf{A}_t = \mathbf{A}_{t-1} - \underbrace{\left(1 - \frac{1}{\|\mathbf{A}_{t-1}\boldsymbol{x}_t\|_2}\right)}_{\text{scalar}} \underbrace{\left(\frac{\mathbf{A}_{t-1}\boldsymbol{x}_t}{\|\mathbf{A}_{t-1}\boldsymbol{x}_t\|_2}\right)}_{\text{vector}} \underbrace{\left(\frac{\mathbf{A}_{t-1}\boldsymbol{x}_t}{\|\mathbf{A}_{t-1}\boldsymbol{x}_t\|_2}\right)^T}_{\text{vector}} \mathbf{A}_{t-1}$$

Table of contents

Convex hull coresets

Desired approximation guarantee

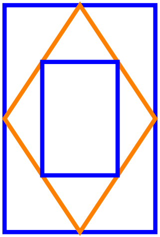
Find a subset S of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ such that there exists a center $\boldsymbol{c}_m \in \mathbb{R}^n$ for which

$$\begin{aligned} & \boldsymbol{c}_m + \operatorname{conv}(X_S) \\ \subseteq\ & \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) \\ \subseteq\ & \boldsymbol{c}_m + \alpha_m \cdot (\operatorname{conv}(X_S) - \boldsymbol{c}_m) \end{aligned}$$

Desired approximation guarantee

Find a subset S of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ such that there exists a center $\boldsymbol{c}_m \in \mathbb{R}^n$ for which

$$\begin{aligned} & \boldsymbol{c}_m + \operatorname{conv}(X_S) \\ \subseteq\ & \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) \\ \subseteq\ & \boldsymbol{c}_m + \alpha_m \cdot (\operatorname{conv}(X_S) - \boldsymbol{c}_m) \end{aligned}$$

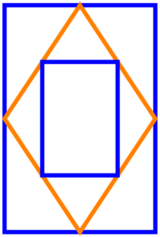


Convex hull coresets

Desired approximation guarantee

Find a subset S of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ such that there exists a center $\boldsymbol{c}_m \in \mathbb{R}^n$ for which

$$\begin{aligned} & \boldsymbol{c}_m + \operatorname{conv}(X_S) \\ \subseteq\ & \operatorname{conv}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) \\ \subseteq\ & \boldsymbol{c}_m + \alpha_m \cdot (\operatorname{conv}(X_S) - \boldsymbol{c}_m) \end{aligned}$$



From ellipsoids to convex hull coresets (informal)

There exists an algorithm that calls the ellipsoidal approximation algorithm as a subroutine and chooses S in an online fashion such that:

$$\alpha_m \leq Cn \log\left(n\kappa^{\mathrm{OL}}(X)\right) \qquad \text{(asymmetric)}$$

$$\alpha_m \leq C\sqrt{n \log\left(n\kappa^{\mathrm{OL}}(X)\right)} \qquad \text{(symmetric)}$$

and $|S| \leq Cn \log\left(n\kappa^{\mathrm{OL}}(X)\right)$.

Conclusion – Streaming Algorithms for Ellipsoidal Approximation of Convex Polytopes

1. We gave **simple, nearly-optimal streaming algorithms** to calculate an ellipsoidal approximation for a convex polytope.

Conclusion – Streaming Algorithms for Ellipsoidal Approximation of Convex Polytopes

1. We gave **simple, nearly-optimal streaming algorithms** to calculate an ellipsoidal approximation for a convex polytope.
2. Our algorithms' space complexities are independent of the length of the stream.

Conclusion – Streaming Algorithms for Ellipsoidal Approximation of Convex Polytopes

1. We gave **simple, nearly-optimal streaming algorithms** to calculate an ellipsoidal approximation for a convex polytope.
2. Our algorithms' space complexities are independent of the length of the stream.
3. We used ellipsoidal approximation algorithms to construct convex hull coresets in a stream.

Conclusion – Streaming Algorithms for Ellipsoidal Approximation of Convex Polytopes

1. We gave **simple, nearly-optimal streaming algorithms** to calculate an ellipsoidal approximation for a convex polytope.
2. Our algorithms' space complexities are independent of the length of the stream.
3. We used ellipsoidal approximation algorithms to construct convex hull coresets in a stream.
4. Papers – https://arxiv.org/abs/2206.07250 (symmetric) and https://arxiv.org/abs/2311.09460 (asymmetric).

Approximating matrix block norms

https://arxiv.org/abs/2311.10013 (in progress)

NSM, Max Ovsiankin

Table of contents

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Approximating matrix block norms

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

$$\mathbf{A} = \overbrace{\begin{bmatrix} \mathbf{A}_1 \in \mathbb{R}^{|S_1|\times n} \\ \hline \mathbf{A}_2 \in \mathbb{R}^{|S_2|\times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3|\times n} \\ \hline \dots \\ \hline \mathbf{A}_m \in \mathbb{R}^{|S_m|\times n} \end{bmatrix}}^{\mathbf{A}}$$

Approximating matrix block norms

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \in \mathbb{R}^{|S_1| \times n} \\ \hline \mathbf{A}_2 \in \mathbb{R}^{|S_2| \times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3| \times n} \\ \hline \dots \\ \hline \mathbf{A}_m \in \mathbb{R}^{|S_m| \times n} \end{bmatrix}$$

$\boldsymbol{x} \in \mathbb{R}^n$

Approximating matrix block norms

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

$$\mathbf{A} = \overbrace{\begin{bmatrix} \mathbf{A}_1 \in \mathbb{R}^{|S_1|\times n} \\ \hline \mathbf{A}_2 \in \mathbb{R}^{|S_2|\times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3|\times n} \\ \hline \dots \\ \hline \mathbf{A}_m \in \mathbb{R}^{|S_m|\times n} \end{bmatrix}}^{\mathbf{A}}$$

$$\boldsymbol{x} \in \mathbb{R}^n \to \underbrace{\begin{bmatrix} \|\mathbf{A}_{S_1}\boldsymbol{x}\|_{p_1} & \|\mathbf{A}_{S_2}\boldsymbol{x}\|_{p_2} & \dots & \|\mathbf{A}_{S_m}\boldsymbol{x}\|_{p_m} \end{bmatrix}}_{\boldsymbol{v}(\boldsymbol{x})}$$

Approximating matrix block norms

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

$$\mathbf{A} = \overbrace{\begin{bmatrix} \mathbf{A}_1 \in \mathbb{R}^{|S_1| \times n} \\ \hline \mathbf{A}_2 \in \mathbb{R}^{|S_2| \times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3| \times n} \\ \hline \dots \\ \hline \mathbf{A}_m \in \mathbb{R}^{|S_m| \times n} \end{bmatrix}}^{\mathbf{A}}$$

$$\boldsymbol{x} \in \mathbb{R}^n \to \underbrace{\begin{bmatrix} \|\mathbf{A}_{S_1}\boldsymbol{x}\|_{p_1} & \|\mathbf{A}_{S_2}\boldsymbol{x}\|_{p_2} & \dots & \|\mathbf{A}_{S_m}\boldsymbol{x}\|_{p_m} \end{bmatrix}}_{\boldsymbol{v}(\boldsymbol{x})} \to \|\boldsymbol{v}(\boldsymbol{x})\|_p^p$$

Approximating matrix block norms

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Find weights $\beta_1, \ldots, \beta_m$, most of which are 0, such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \sum_{i=1}^{m} \beta_i \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

$$\overbrace{\begin{bmatrix} \mathbf{A}_1 \in \mathbb{R}^{|S_1| \times n} \\ \hline \mathbf{A}_2 \in \mathbb{R}^{|S_2| \times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3| \times n} \\ \hline \ldots \\ \hline \mathbf{A}_m \in \mathbb{R}^{|S_m| \times n} \end{bmatrix}}^{\mathbf{A}}$$

$$\boldsymbol{x} \in \mathbb{R}^n \to \underbrace{\begin{bmatrix} \|\mathbf{A}_{S_1}\boldsymbol{x}\|_{p_1} & \|\mathbf{A}_{S_2}\boldsymbol{x}\|_{p_2} & \ldots & \|\mathbf{A}_{S_m}\boldsymbol{x}\|_{p_m} \end{bmatrix}}_{\boldsymbol{v}(\boldsymbol{x})} \to \|\boldsymbol{v}(\boldsymbol{x})\|_p^p$$

Approximating matrix block norms

Matrix block norm sparsification

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Find weights $\beta_1, \ldots, \beta_m$, most of which are 0, such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \sum_{i=1}^{m} \beta_i \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

$$\overbrace{\begin{bmatrix} \mathbf{A}_1 \in \mathbb{R}^{|S_1| \times n} \\ \hline \mathbf{A}_2 \in \mathbb{R}^{|S_2| \times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3| \times n} \\ \hline \cdots \\ \hline \mathbf{A}_m \in \mathbb{R}^{|S_m| \times n} \end{bmatrix}}^{\mathbf{A}} \to \overbrace{\begin{bmatrix} \beta_1\mathbf{A}_1 \in \mathbb{R}^{|S_1| \times n} \\ \hline \beta_2\mathbf{A}_2 \in \mathbb{R}^{|S_2| \times n} \\ \hline \mathbf{A}_3 \in \mathbb{R}^{|S_3| \times n} \\ \hline \cdots \\ \hline \beta_m\mathbf{A}_m \in \mathbb{R}^{|S_m| \times n} \end{bmatrix}}^{\mathbf{A}'}$$

$$\boldsymbol{x} \in \mathbb{R}^n \to \underbrace{\begin{bmatrix} \|\mathbf{A}_{S_1}\boldsymbol{x}\|_{p_1} & \|\mathbf{A}_{S_2}\boldsymbol{x}\|_{p_2} & \ldots & \|\mathbf{A}_{S_m}\boldsymbol{x}\|_{p_m} \end{bmatrix}}_{\boldsymbol{v}(\boldsymbol{x})} \to \|\boldsymbol{v}(\boldsymbol{x})\|_p^p$$

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.
- $p = 1$ and $p_1 = \cdots = p_m = 2 \to \ell_{1,2}$ norm (used for feature selection).

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.
- $p = 1$ and $p_1 = \cdots = p_m = 2 \to \ell_{1,2}$ norm (used for feature selection).
- Recovers matrix p, q norm:

$$\|\mathbf{X}\|_{p,q}$$

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.
- $p = 1$ and $p_1 = \cdots = p_m = 2 \to \ell_{1,2}$ norm (used for feature selection).
- Recovers matrix p, q norm:

$$\|\mathbf{X}\|_{p,q} = \left\| \begin{bmatrix} \boldsymbol{x}_1 \in \mathbb{R}^r \\ \hline \boldsymbol{x}_2 \in \mathbb{R}^r \\ \hline \dots \\ \hline \boldsymbol{x}_m \in \mathbb{R}^r \end{bmatrix} \right\|_{p,q}$$

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.
- $p = 1$ and $p_1 = \cdots = p_m = 2 \to \ell_{1,2}$ norm (used for feature selection).
- Recovers matrix p, q norm:

$$\|\mathbf{X}\|_{p,q} = \left\| \begin{bmatrix} \boldsymbol{x}_1 \in \mathbb{R}^r \\ \hline \boldsymbol{x}_2 \in \mathbb{R}^r \\ \hline \dots \\ \hline \boldsymbol{x}_m \in \mathbb{R}^r \end{bmatrix} \right\|_{p,q} = \left\| \begin{bmatrix} \|\boldsymbol{x}_1\|_q \\ \hline \|\boldsymbol{x}_2\|_q \\ \hline \dots \\ \hline \|\boldsymbol{x}_m\|_q \end{bmatrix} \right\|_p$$

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p .$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.
- $p = 1$ and $p_1 = \cdots = p_m = 2 \to \ell_{1,2}$ norm (used for feature selection).
- Recovers matrix p, q norm:

$$\|\mathbf{X}\|_{p,q} = \left\| \begin{bmatrix} \boldsymbol{x}_1 \in \mathbb{R}^r \\ \hline \boldsymbol{x}_2 \in \mathbb{R}^r \\ \hline \dots \\ \hline \boldsymbol{x}_m \in \mathbb{R}^r \end{bmatrix} \right\|_{p,q} = \left\| \begin{bmatrix} \|\boldsymbol{x}_1\|_q \\ \hline \|\boldsymbol{x}_2\|_q \\ \hline \dots \\ \hline \|\boldsymbol{x}_m\|_q \end{bmatrix} \right\|_{p} = \left(\sum_{i=1}^{m} \|\boldsymbol{x}_i\|_q^p \right)^{1/p}$$

Motivation

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p.$$

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m}\times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

- $p = p_1 = \cdots = p_m \to$ higher-rank analog of row sampling for p-norms.
- $p = 2$ and $p_1 = \cdots = p_m = \infty \to$ hypergraph Laplacian.
- $p = 1$ and $p_1 = \cdots = p_m = 2 \to \ell_{1,2}$ norm (used for feature selection).
- Recovers matrix p, q norm:

$$\|\mathbf{X}\|_{p,q} = \left\| \begin{bmatrix} \boldsymbol{x}_1 \in \mathbb{R}^r \\ \hline \boldsymbol{x}_2 \in \mathbb{R}^r \\ \hline \dots \\ \hline \boldsymbol{x}_m \in \mathbb{R}^r \end{bmatrix} \right\|_{p,q} = \left\| \begin{bmatrix} \|\boldsymbol{x}_1\|_q \\ \hline \|\boldsymbol{x}_2\|_q \\ \hline \dots \\ \hline \|\boldsymbol{x}_m\|_q \end{bmatrix} \right\|_{p} = \left(\sum_{i=1}^{m} \|\boldsymbol{x}_i\|_q^p \right)^{1/p}$$

Applications – subroutine to speed up regression, dataset summarization, etc

Table of contents

Our results

Regimes of p and $p_1, \ldots, p_m$:

- $p > 0$ and $p_1, \ldots, p_m = p$;
- $p \geq 1$ and $p_1, \ldots, p_m \geq 2$.

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m} \times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \left\|\mathbf{A}'\boldsymbol{x}\right\|_{\mathcal{G}_p}^p$$

Our results

Regimes of p and $p_1, \ldots, p_m$:

- $p > 0$ and $p_1, \ldots, p_m = p$;
- $p \geq 1$ and $p_1, \ldots, p_m \geq 2$.

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m} \times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \left\|\mathbf{A}'\boldsymbol{x}\right\|_{\mathcal{G}_p}^p$$

Theorem

1. (Existence) If p and p_i belong to at least one of the regimes above, then there exists a weight vector $\beta \in \mathbb{R}^m_{\geq 0}$ such that

$$\widetilde{m} := \|\beta\|_0 = C(p, p_1, \ldots, p_m) \cdot \frac{n^{\max(1, p/2)} \log\left(n/\varepsilon\right) (\log n)^2}{\varepsilon^2}.$$

Our results

Regimes of p and $p_1, \ldots, p_m$:

- $p > 0$ and $p_1, \ldots, p_m = p$;
- $p \geq 1$ and $p_1, \ldots, p_m \geq 2$.

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m} \times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

Theorem

1. (Existence) If p and p_i belong to at least one of the regimes above, then there exists a weight vector $\beta \in \mathbb{R}^m_{\geq 0}$ such that

$$\widetilde{m} := \|\beta\|_0 = C(p, p_1, \ldots, p_m) \cdot \frac{n^{\max(1,p/2)} \log(n/\varepsilon) (\log n)^2}{\varepsilon^2}.$$

Our results

Regimes of p and $p_1, \ldots, p_m$:

- $p > 0$ and $p_1, \ldots, p_m = p$;
- $p \geq 1$ and $p_1, \ldots, p_m \geq 2$.

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m} \times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

Theorem

1. (Existence) If p and p_i belong to at least one of the regimes above, then there exists a weight vector $\beta \in \mathbb{R}^m_{\geq 0}$ such that

$$\widetilde{m} := \|\beta\|_0 = C(p, p_1, \ldots, p_m) \cdot \frac{n^{\max(1, p/2)} \log\left(n/\varepsilon\right) (\log n)^2}{\varepsilon^2}.$$

2. (Computation) If $p = p_1 = \cdots = p_m$, or $p_1 = \cdots = p_m = 2$ and $p > 0$, or $p = 2$ and $p_1, \ldots, p_m \geq 2$, then β can be found in polylogarithmically many leverage score computations.

Our results

Regimes of p and $p_1, \ldots, p_m$:

- $p > 0$ and $p_1, \ldots, p_m = p$;
- $p \geq 1$ and $p_1, \ldots, p_m \geq 2$.

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m} \times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

Theorem

1. (Existence) If p and p_i belong to at least one of the regimes above, then there exists a weight vector $\beta \in \mathbb{R}_{\geq 0}^m$ such that

$$\widetilde{m} := \|\beta\|_0 = C(p, p_1, \ldots, p_m) \cdot \frac{n^{\max(1, p/2)} \log\left(n/\varepsilon\right) (\log n)^2}{\varepsilon^2}.$$

2. (Computation) If $p = p_1 = \cdots = p_m$, or $p_1 = \cdots = p_m = 2$ and $p > 0$, or $p = 2$ and $p_1, \ldots, p_m \geq 2$, then β can be found in polylogarithmically many leverage score computations.

Approximate leverage scores ($\boldsymbol{a}_i^T \left(\mathbf{A}^T\mathbf{D}\mathbf{A}\right)^{-1} \boldsymbol{a}_i$ for nonnegative diagonal $\mathbf{D}$) can be found in time $\widetilde{O}(\mathrm{nnz}(\mathbf{A}) + n^\omega)$.

Our results

Regimes of p and $p_1, \ldots, p_m$:

- $p > 0$ and $p_1, \ldots, p_m = p$;
- $p \geq 1$ and $p_1, \ldots, p_m \geq 2$.

Want a weighted subset of rows $\mathbf{A}' \in \mathbb{R}^{\widetilde{m} \times n}$ such that

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \in (1 \pm \varepsilon) \cdot \|\mathbf{A}'\boldsymbol{x}\|_{\mathcal{G}_p}^p$$

Theorem

1. (Existence) If p and p_i belong to at least one of the regimes above, then there exists a weight vector $\beta \in \mathbb{R}^m_{\geq 0}$ such that

$$\widetilde{m} := \|\beta\|_0 = C(p, p_1, \ldots, p_m) \cdot \frac{n^{\max(1,p/2)} \log\left(n/\varepsilon\right) (\log n)^2}{\varepsilon^2}.$$

2. (Computation) If $p = p_1 = \cdots = p_m$, or $p_1 = \cdots = p_m = 2$ and $p > 0$, or $p = 2$ and $p_1, \ldots, p_m \geq 2$, then β can be found in polylogarithmically many leverage score computations.

Approximate leverage scores ($\boldsymbol{a}_i^T \left(\mathbf{A}^T\mathbf{D}\mathbf{A}\right)^{-1} \boldsymbol{a}_i$ for nonnegative diagonal $\mathbf{D}$) can be found in time $\widetilde{O}(\mathrm{nnz}(\mathbf{A}) + n^\omega)$.

Dependence on n is essentially optimal (we need $\widetilde{m} \gtrsim n^{\max(1,p/2)}$ [LWW19]).

Table of contents

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\boldsymbol{\rho}_i} \cdot \left\|\mathbf{A}_{S_i}\boldsymbol{x}\right\|_{p_i}^p$ with probability $\boldsymbol{\rho}_i$.

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\boldsymbol{\rho}_i} \cdot \left\|\mathbf{A}_{S_i}\boldsymbol{x}\right\|_{p_i}^p$ with probability $\boldsymbol{\rho}_i$.
4. Return $\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})$.

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\boldsymbol{\rho}_i} \cdot \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$ with probability $\boldsymbol{\rho}_i$.
4. Return $\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})$.

This estimator is unbiased for all $\boldsymbol{x} \in \mathbb{R}^n$:

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\rho_1, \ldots, \rho_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\rho_i} \cdot \left\|\mathbf{A}_{S_i}\boldsymbol{x}\right\|_{p_i}^p$ with probability ρ_i.
4. Return $\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})$.

This estimator is unbiased for all $\boldsymbol{x} \in \mathbb{R}^n$:

$$\mathbb{E}\left[\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})\right] = \frac{1}{\widetilde{m}} \mathbb{E}\left[\sum_{h=1}^{\widetilde{m}} \frac{1}{\rho_{i_h}} \cdot \left\|\mathbf{A}_{S_{i_h}}\boldsymbol{x}\right\|_{p_{i_h}}^p\right]$$

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\boldsymbol{\rho}_i} \cdot \left\|\mathbf{A}_{S_i}\boldsymbol{x}\right\|_{p_i}^p$ with probability $\boldsymbol{\rho}_i$.
4. Return $\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})$.

This estimator is unbiased for all $\boldsymbol{x} \in \mathbb{R}^n$:

$$\mathbb{E}\left[\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})\right] = \frac{1}{\widetilde{m}}\mathbb{E}\left[\sum_{h=1}^{\widetilde{m}} \frac{1}{\boldsymbol{\rho}_{i_h}} \cdot \left\|\mathbf{A}_{S_{i_h}}\boldsymbol{x}\right\|_{p_{i_h}}^p\right] = \frac{1}{\widetilde{m}} \cdot \left(\widetilde{m}\, \|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p\right)$$

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\boldsymbol{\rho}_i} \cdot \left\|\mathbf{A}_{S_i}\boldsymbol{x}\right\|_{p_i}^p$ with probability $\boldsymbol{\rho}_i$.
4. Return $\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})$.

This estimator is unbiased for all $\boldsymbol{x} \in \mathbb{R}^n$:

$$\mathbb{E}\left[\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})\right] = \frac{1}{\widetilde{m}}\mathbb{E}\left[\sum_{h=1}^{\widetilde{m}} \frac{1}{\boldsymbol{\rho}_{i_h}} \cdot \left\|\mathbf{A}_{S_{i_h}}\boldsymbol{x}\right\|_{p_{i_h}}^p\right] = \frac{1}{\widetilde{m}} \cdot \left(\widetilde{m}\,\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p\right) = \|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p .$$

Importance sampling

$$\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p := \sum_{i=1}^{m} \|\mathbf{A}_{S_i}\boldsymbol{x}\|_{p_i}^p$$

Meta algorithm

1. Let $f(\boldsymbol{x}) = 0$.
2. Identify a probability distribution $\mathcal{D} = (\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_m)$.
3. Repeat $\widetilde{m}$ times:
   3.1 Update $f(\boldsymbol{x}) \leftarrow f(\boldsymbol{x}) + \frac{1}{\boldsymbol{\rho}_i} \cdot \left\|\mathbf{A}_{S_i}\boldsymbol{x}\right\|_{p_i}^p$ with probability $\boldsymbol{\rho}_i$.
4. Return $\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})$.

This estimator is unbiased for all $\boldsymbol{x} \in \mathbb{R}^n$:

$$\mathbb{E}\left[\frac{1}{\widetilde{m}} \cdot f(\boldsymbol{x})\right] = \frac{1}{\widetilde{m}}\mathbb{E}\left[\sum_{h=1}^{\widetilde{m}} \frac{1}{\boldsymbol{\rho}_{i_h}} \cdot \left\|\mathbf{A}_{S_{i_h}}\boldsymbol{x}\right\|_{p_{i_h}}^p\right] = \frac{1}{\widetilde{m}} \cdot \left(\widetilde{m}\,\|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p\right) = \|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p .$$

Our task: Find $\mathcal{D}$ such that for a "small" $\widetilde{m}$, w.h.p, for all $\boldsymbol{x} \in \mathbb{R}^n$:

$$(1-\varepsilon) \cdot \frac{f(\boldsymbol{x})}{\widetilde{m}} \le \|\mathbf{A}\boldsymbol{x}\|_{\mathcal{G}_p}^p \le (1+\varepsilon) \cdot \frac{f(\boldsymbol{x})}{\widetilde{m}}.$$

Conclusion – approximating matrix block norms

1. We gave existence and algorithmic results for finding sparse approximations to matrix block norms.

Conclusion – approximating matrix block norms

1. We gave existence and algorithmic results for finding sparse approximations to matrix block norms.
2. Our existence results yield sparsities whose sizes are nearly optimal in the input dimension n.

Conclusion – approximating matrix block norms

1. We gave existence and algorithmic results for finding sparse approximations to matrix block norms.
2. Our existence results yield sparsities whose sizes are nearly optimal in the input dimension n.
3. The algorithmic results when $p_1 = \cdots = p_m = p > 0$, or when $p > 0$ and $p_1 = \cdots = p_m = 2$, or when $p = 2$ and $p_1, \ldots, p_m \geq 2$, run in time $\widetilde{O}(\mathrm{nnz}(\mathbf{A}) + n^{\omega})$.

Conclusion – approximating matrix block norms

1. We gave existence and algorithmic results for finding sparse approximations to matrix block norms.
2. Our existence results yield sparsities whose sizes are nearly optimal in the input dimension n.
3. The algorithmic results when $p_1 = \cdots = p_m = p > 0$, or when $p > 0$ and $p_1 = \cdots = p_m = 2$, or when $p = 2$ and $p_1, \ldots, p_m \geq 2$, run in time $\widetilde{O}(\mathrm{nnz}(\mathbf{A}) + n^{\omega})$.
4. Paper – https://arxiv.org/abs/2311.10013.

Modern input concerns for algorithmic data science

The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

Unwieldy data

Modern input concerns for algorithmic data science

The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]

Unwieldy data

Modern input concerns for algorithmic data science

The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]
- Preference based optimization with a monotone adversary [BGLMSY24]

Unwieldy data

Modern input concerns for algorithmic data science

The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]
- Preference based optimization with a monotone adversary [BGLMSY24]
- Generalization of short program interpolators [MS23]

Unwieldy data

Modern input concerns for algorithmic data science

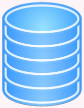
The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]
- Preference based optimization with a monotone adversary [BGLMSY24]
- Generalization of short program interpolators [MS23]


Unwieldy data

- Streaming ellipsoidal approximations of convex polytopes [MMO22; MMO24]



Modern input concerns for algorithmic data science

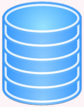
The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]
- Preference based optimization with a monotone adversary [BGLMSY24]
- Generalization of short program interpolators [MS23]

Unwieldy data

- Streaming ellipsoidal approximations of convex polytopes [MMO22; MMO24]
- Sparse approximations of matrix block norms [MO23]



Modern input concerns for algorithmic data science

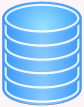
The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]
- Preference based optimization with a monotone adversary [BGLMSY24]
- Generalization of short program interpolators [MS23]

Unwieldy data

- Streaming ellipsoidal approximations of convex polytopes [MMO22; MMO24]
- Sparse approximations of matrix block norms [MO23]
- Finding short certificates for monotone functions [GM22]



Modern input concerns for algorithmic data science

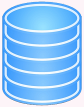
The input in a data science problem instance isn't always clean or accessible.

Untrustworthy and unexpected data

- Robustness of ML to train time corruptions [MB21]
- Preference based optimization with a monotone adversary [BGLMSY24]
- Generalization of short program interpolators [MS23]

Unwieldy data

- **Streaming ellipsoidal approximations of convex polytopes [MMO22; MMO24]**
- **Sparse approximations of matrix block norms [MO23]**
- Finding short certificates for monotone functions [GM22]



Questions?

Thank you!!

References I

Avrim Blum, Meghal Gupta, Gene Li, Naren Sarayu Manoj, Aadirupa Saha, and Yuanyuan Yang. Dueling optimization with a monotone adversary. In *Proceedings of Thirty Fifth Conference on Algorithmic Learning Theory (ALT)*, February 2024 (cited on pages 102–109).

Michael B. Cohen, Ben Cousins, Yin Tat Lee, and Xin Yang. A near-optimal algorithm for approximating the John ellipsoid. In *Proceedings of the Conference on Learning Theory*, volume 99, pages 849–873, 2019 (cited on pages 12–17).

Meghal Gupta and Naren Sarayu Manoj. An optimal algorithm for certifying monotone functions. *arXiv preprint arXiv:2204.01224*, 2022 (cited on pages 102–109).

Fritz John. Extremum problems with inequalities as subsidiary conditions. In *Studies and Essays Presented to R. Courant on his 60th Birthday*, pages 187–204. Interscience Publishers, Inc, 1948 (cited on pages 12–17).

References II

Yi Li, Ruosong Wang, and David P. Woodruff. Tight bounds for the subspace sketch problem with applications. 2019. DOI: 10.48550/arXiv.1904.05543. URL: https://arxiv.org/abs/1904.05543 (cited on pages 79–84).

Yury Makarychev, Naren Sarayu Manoj, and Max Ovsiankin. Streaming algorithms for ellipsoidal approximation of convex polytopes. In *Proceedings of Thirty Fifth Conference on Learning Theory (COLT)*, pages 3070–3093, July 2022 (cited on pages 106–109).

Yury Makarychev, Naren Sarayu Manoj, and Max Ovsiankin. Near-optimal streaming ellipsoidal rounding for general convex polytopes. In *Proceedings of Fifty Sixth Annual ACM Symposium on Theory of Computing (STOC)*, June 2024 (cited on pages 106–109).

Naren Sarayu Manoj and Avrim Blum. Excess capacity and backdoor poisoning. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 2021 (cited on pages 102–109).

References III

Naren Sarayu Manoj and Max Ovsiankin. The change-of-measure method, block lewis weights, and approximating matrix block norms. 2023. arXiv: 2311.10013 [math.FA] (cited on pages 107–109).

Naren Sarayu Manoj and Nathan Srebro. Shortest program interpolation learning. In *Proceedings of Thirty Sixth Conference on Learning Theory (COLT)*, pages 4881–4901, July 2023 (cited on pages 102–109).