Scalable and Robust Nonconvex Approaches for Low-rank Structure Estimation

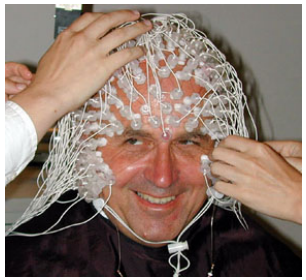
Yuejie Chi

Carnegie Mellon University

International Workshop on Intelligent Signal Processing
Zhejiang University, September 2021

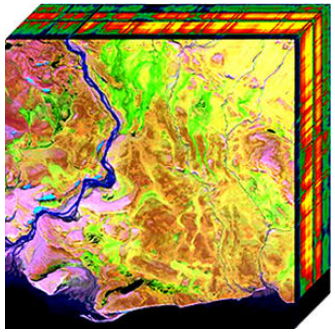
Sensing and imaging advances

New imaging/sensing modalities allow us to probe the nature in unprecedented manners.

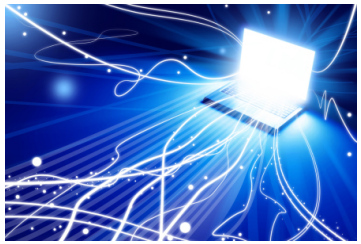


healthcare

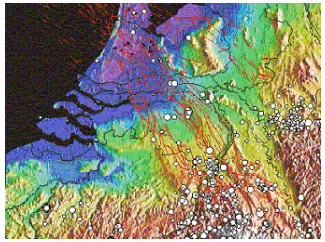
Radio astronomy



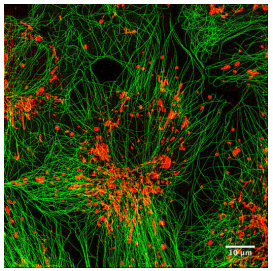
hyperspectral



Internet traffic



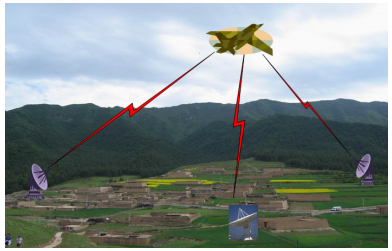
seismic imaging



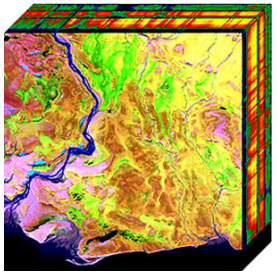
microscopy

The large amount of data brings exciting opportunities that call for new tools that are **scalable in computation and memory**.

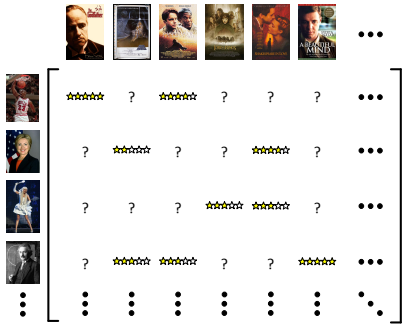
Low-rank matrices in data science



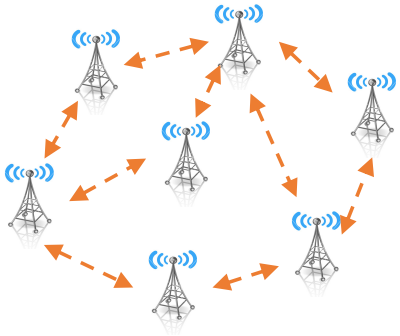
radar imaging



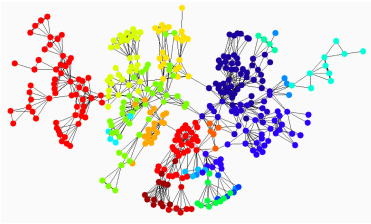
hyperspectral imaging



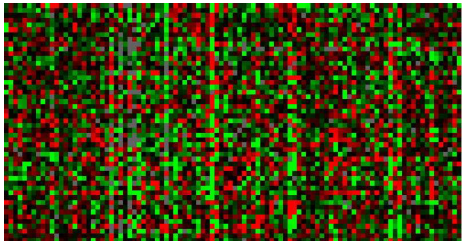
recommendation systems



localization



community detection



bioinformatics

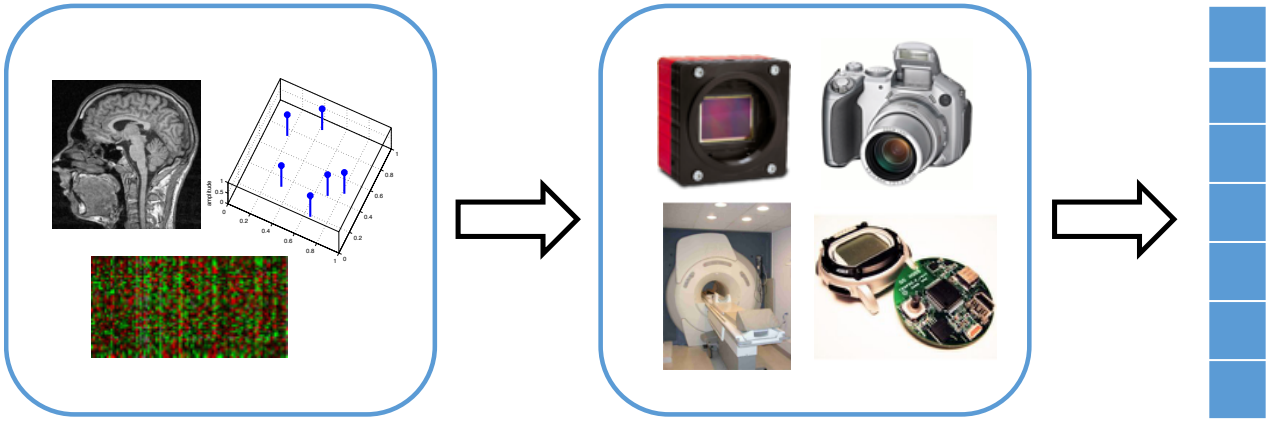
Low-rank matrices are redundant representations of latent information

Low-rank matrix sensing

$\boldsymbol{M} \in \mathbb{R}^{n_1 \times n_2}$
$\mathrm{rank}(\boldsymbol{M}) = r$

$\mathcal{A}(\cdot)$
linear map

$\boldsymbol{y} \in \mathbb{R}^m$



$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{M}) + \text{noise}$$

Recover M in the sample-starved regime:

$$\underbrace{(n_1 + n_2)r}_{\text{degree of freedom}} \lesssim \underbrace{m}_{\text{sensing budget}} \ll \underbrace{n_1 n_2}_{\text{ambient dimension}}$$

Convex relaxation via nuclear norm minimization

$$\min_{\boldsymbol{Z} \in \mathbb{R}^{n_1 \times n_2}} \quad \mathsf{rank}(\boldsymbol{Z}) \qquad\qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

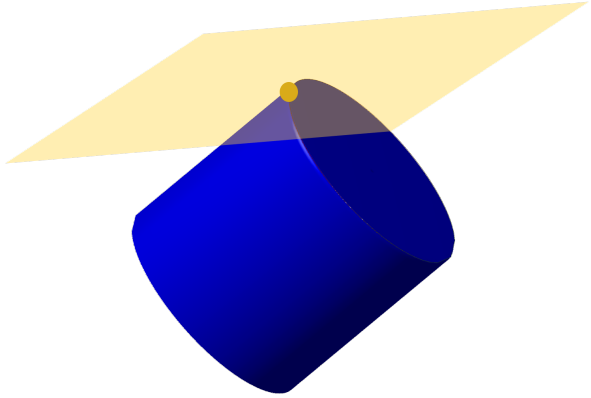
Convex relaxation via nuclear norm minimization

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \text{rank}(\boldsymbol{Z}) \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

$\Downarrow$ cvx surrogate

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \|\boldsymbol{Z}\|_* \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

where $\|\cdot\|_*$ is the nuclear norm.



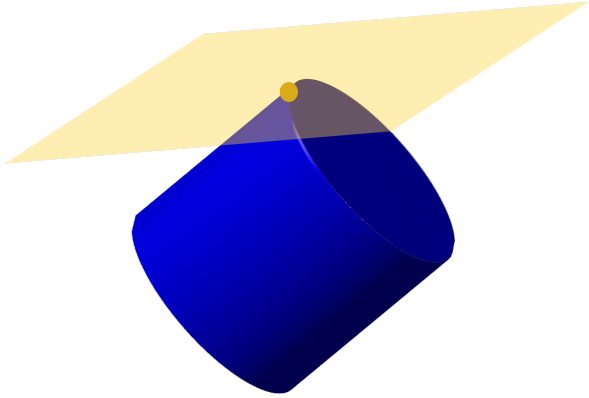
Convex relaxation via nuclear norm minimization

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \text{rank}(\boldsymbol{Z}) \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

$\Downarrow$ cvx surrogate

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \|\boldsymbol{Z}\|_* \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

where $\|\cdot\|_*$ is the nuclear norm.



Significant developments in the last decade:

Fazel '02, Recht, Parrilo, Fazel '10, Candès, Recht '09, Candès, Tao '10, Cai et al. '10, Gross '10, Negahban, Wainwright '11, Sanghavi et al. '13, Chen, Chi '14, ...

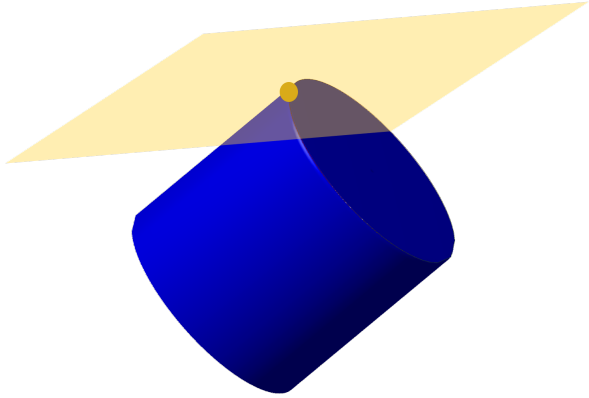
Convex relaxation via nuclear norm minimization

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \text{rank}(\boldsymbol{Z}) \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

$\Downarrow$ cvx surrogate

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \|\boldsymbol{Z}\|_* \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

where $\|\cdot\|_*$ is the nuclear norm.



Significant developments in the last decade:

Fazel '02, Recht, Parrilo, Fazel '10, Candès, Recht '09, Candès, Tao '10, Cai et al. '10, Gross '10, Negahban, Wainwright '11, Sanghavi et al. '13, Chen, Chi '14, ...

Poor scalability: operate in the *ambient* matrix space

Low-rank matrix factorization

$$\min_{\boldsymbol{Z} \in \mathbb{R}^{n_1 \times n_2}} \quad \mathsf{rank}(\boldsymbol{Z}) \qquad\qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

Low-rank matrix factorization

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \mathsf{rank}(\boldsymbol{Z}) \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

$$\Downarrow$$

$$\min_{\mathrm{rank}(\boldsymbol{Z})=r} \quad \frac{1}{2}\|\boldsymbol{y}-\mathcal{A}(\boldsymbol{Z})\|_2^2$$

Low-rank matrix factorization

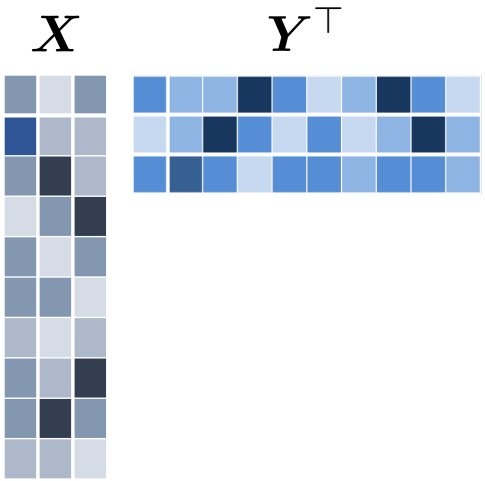
$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \mathsf{rank}(\boldsymbol{Z}) \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

$$\Downarrow$$

$$\min_{\mathrm{rank}(\boldsymbol{Z})=r} \quad \frac{1}{2}\left\|\boldsymbol{y}-\mathcal{A}(\boldsymbol{Z})\right\|_2^2$$

$$\Downarrow$$

$\boldsymbol{Z} =$



$$\min_{\boldsymbol{X}\in\mathbb{R}^{n_1\times r},\boldsymbol{Y}\in\mathbb{R}^{n_2\times r}} \quad f(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\left\|\boldsymbol{y}-\mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top)\right\|_2^2$$

Low-rank matrix factorization

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n_1\times n_2}} \quad \mathrm{rank}(\boldsymbol{Z}) \qquad \text{s.t.} \quad \boldsymbol{y} \approx \mathcal{A}(\boldsymbol{Z})$$

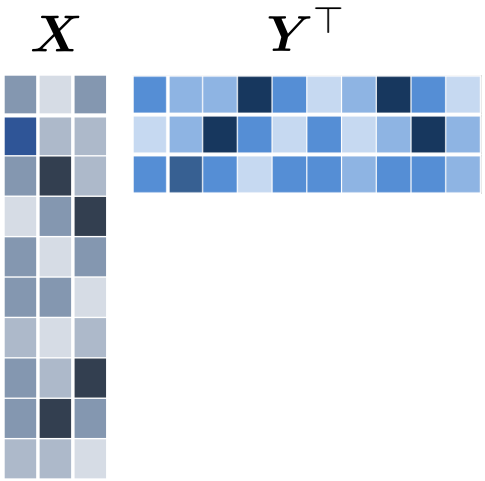
$$\Downarrow$$

$$\min_{\mathrm{rank}(\boldsymbol{Z})=r} \quad \frac{1}{2}\|\boldsymbol{y}-\mathcal{A}(\boldsymbol{Z})\|_2^2$$

more scalable,
but nonconvex!

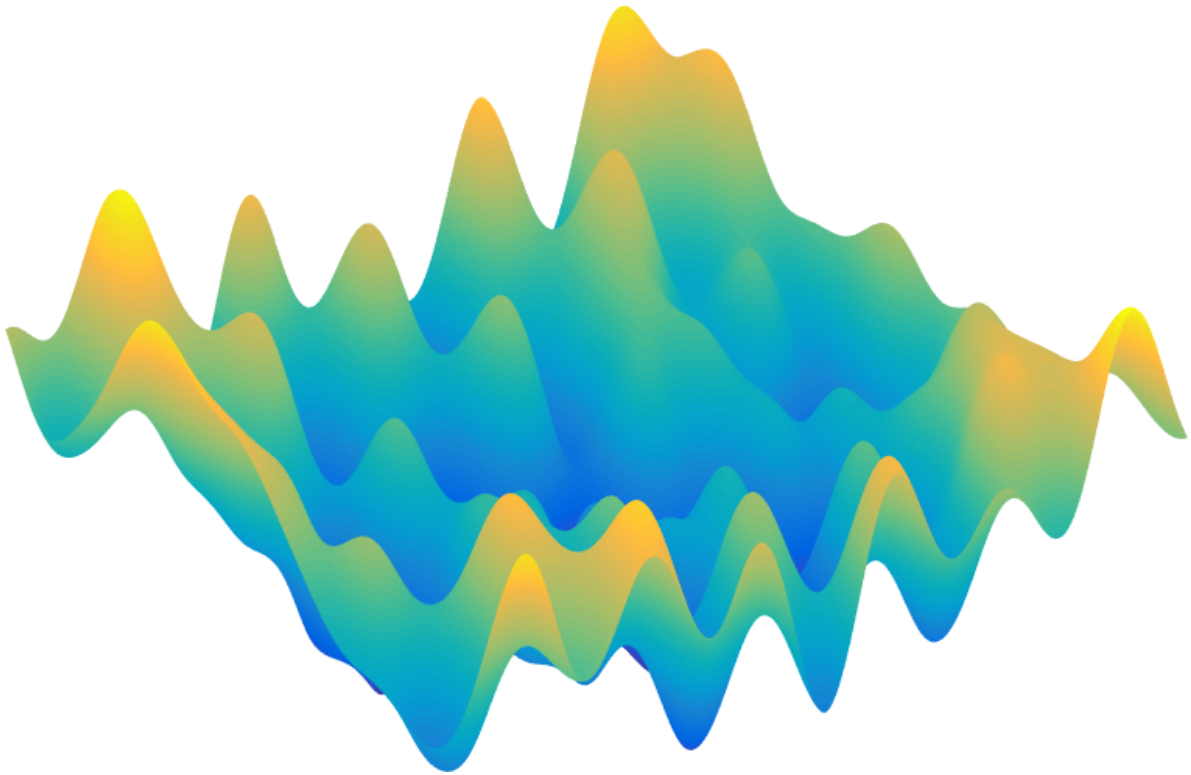
$$\Downarrow$$

$\boldsymbol{Z} =$



$$\min_{\boldsymbol{X}\in\mathbb{R}^{n_1\times r},\boldsymbol{Y}\in\mathbb{R}^{n_2\times r}} \quad f(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\left\|\boldsymbol{y}-\mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top)\right\|_2^2$$

Nonconvex problems are hard (in theory)!



"...in fact, the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

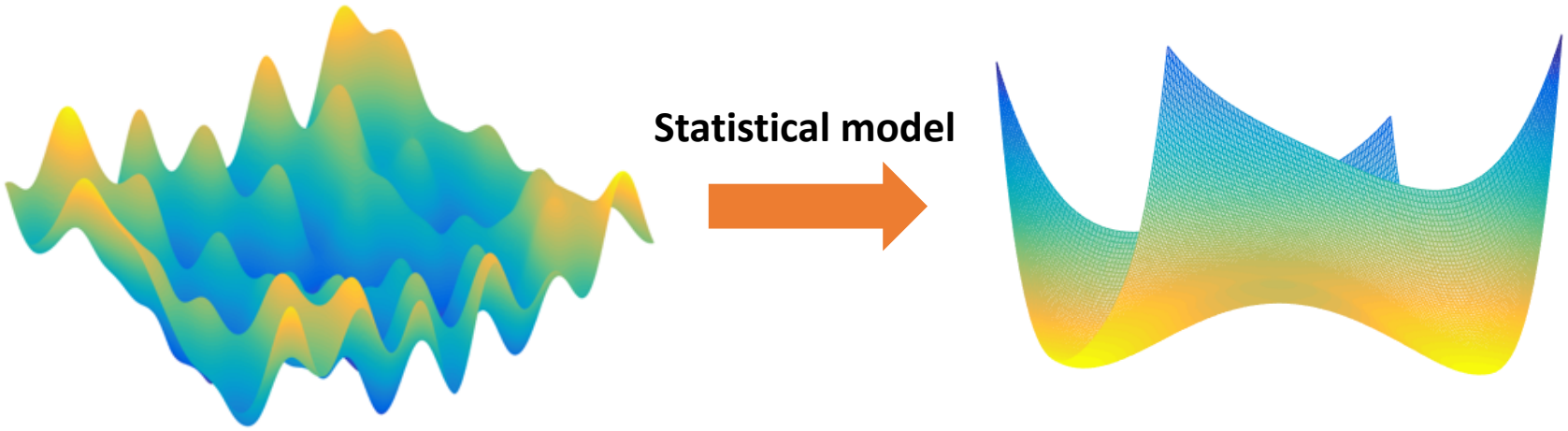
R. T. Rockafellar, in SIAM Review, 1993

Nonconvex problems are hard (in theory)!

"...in fact, the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

R. T. Rockafellar, in SIAM Review, 1993

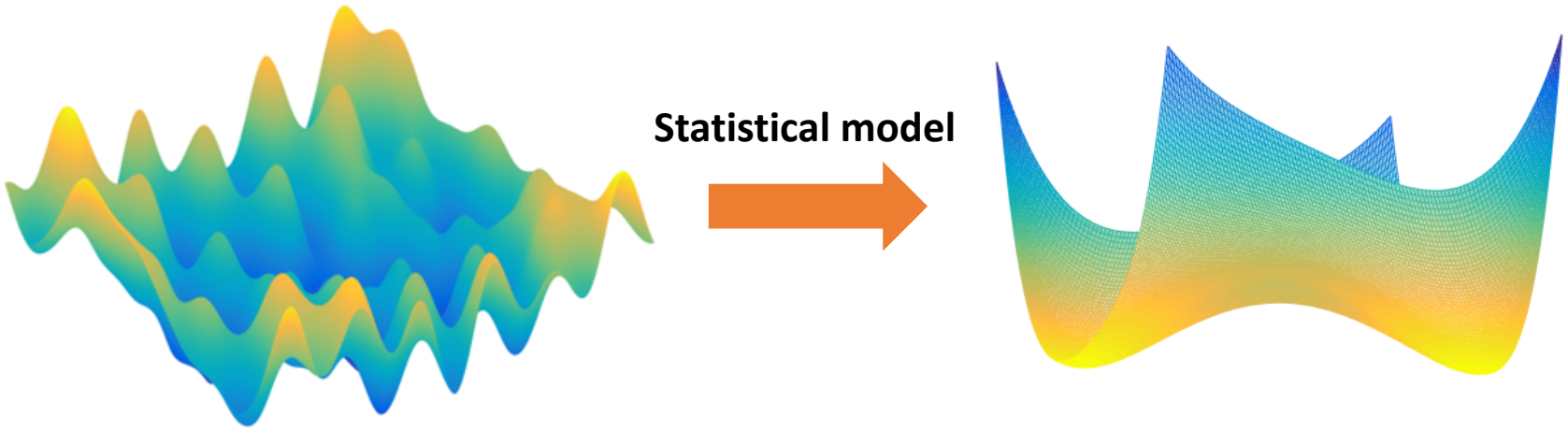
Statistics meets optimization



worst case

average case

Statistics meets optimization

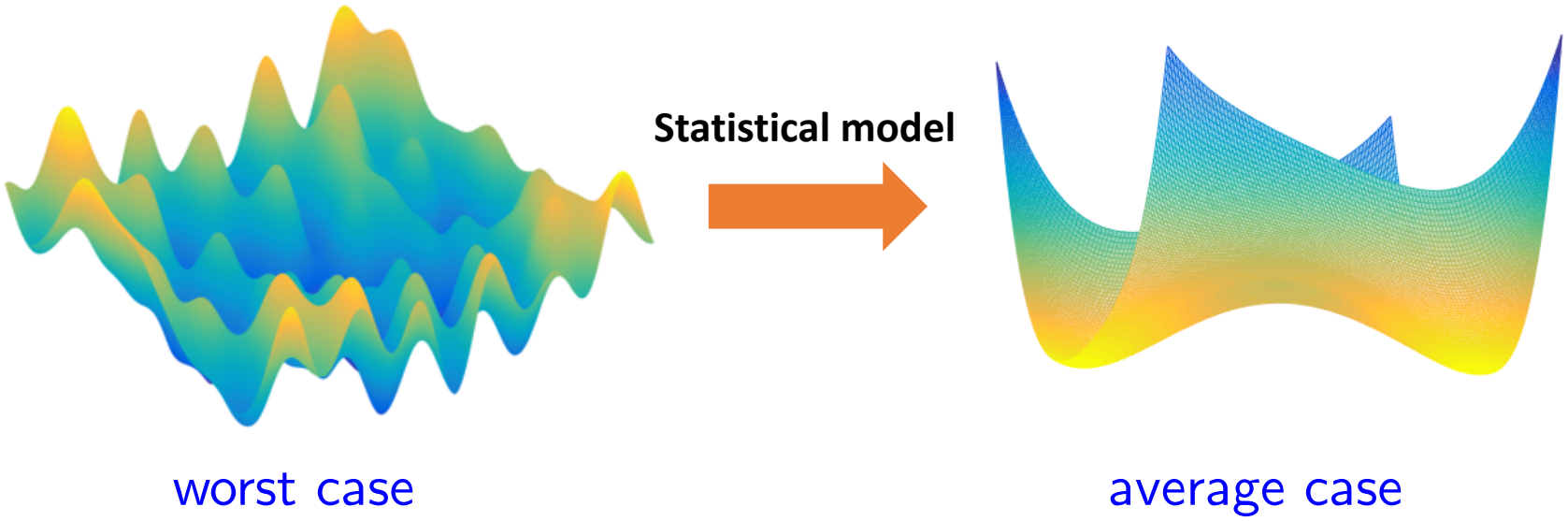


worst case

average case

Simple algorithms can be efficient for nonconvex learning!

Statistics meets optimization



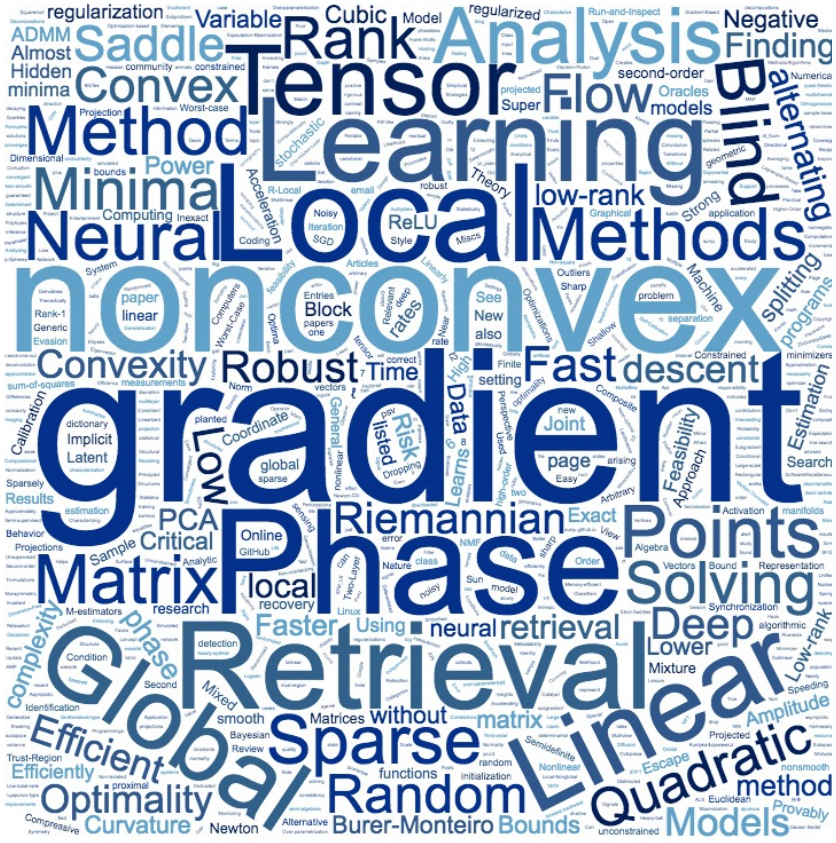
Simple algorithms can be efficient for nonconvex learning!

Vanilla gradient descent (GD):

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla f(\boldsymbol{x}_t)$$

for $t = 0, 1, \ldots$

Recent developments: provable nonconvex optimization



"Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview," Chi, Lu, Chen, TSP 2019

Phase retrieval: Netrapalli et al. '13, Candès, Li, Soltanolkotabi '14, Chen, Candès '15, Cai, Li, Ma '15, Zhang et al. '16, Wang et al. '16, Sun, Qu, Wright '16, Ma et al. '17, Chen et al. '18, Soltani, Hegde '18, Ruan and Duchi, '18, ...

Matrix sensing/completion: Keshavan et al. '09, Jain et al. '09, Hardt '13, Jain et al. '13, Sun, Luo '15, Chen, Wainwright '15, Tu et al. '15, Zheng, Lafferty '15, Bhojanapalli et al. 16, Ge, Lee, Ma '16, Jin et al. '16, Ma et al. '17, Chen and Li '17, Cai et al. '18, Li, Zhu, Tang, Wakin '18, Charisopoulos et al. '19, ...

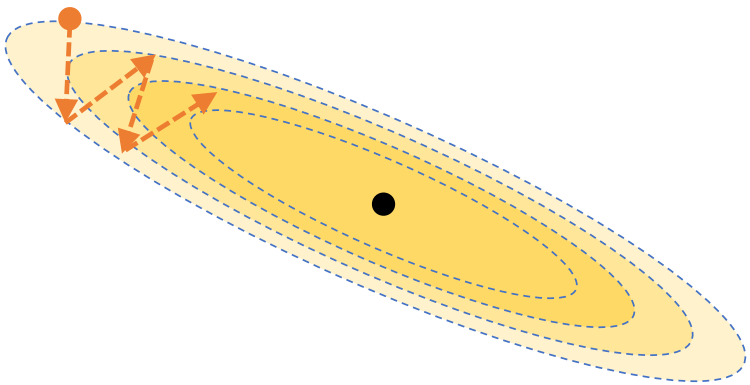
Blind deconvolution/demixing: Li et al. '16, Lee et al. '16, Cambareri, Jacques '16, Ling, Strohmer '16, Huang, Hand '16, Ma et al. '17, Zhang et al. '18, Li, Bresler '18, Dong, Shi '18, Shi, Chi '19, Qu et al. '19...

Dictionary learning: Arora et al. '14, Sun et al. '15, Chatterji, Bartlett '17, Bai et al. '18, Gilboa et al. '18, Rambhatla et al. '19, Qu et al. '19,...

Robust principal component analysis: Netrapalli et al. '14, Yi et al. '16, Gu et al. '16, Ge et al. '17, Cherapanamjeri et al. '17, Vaswani et al. '18, Maunu et al. '19, ...

Deep learning: Zhong et al. '17, Bai, Mei, Montanari '17, Du et al. '17, Ge, Lee, Ma '17, Du et al. '18, Soltanolkotabi and Oymak, '18...

Acceleration via preconditioning

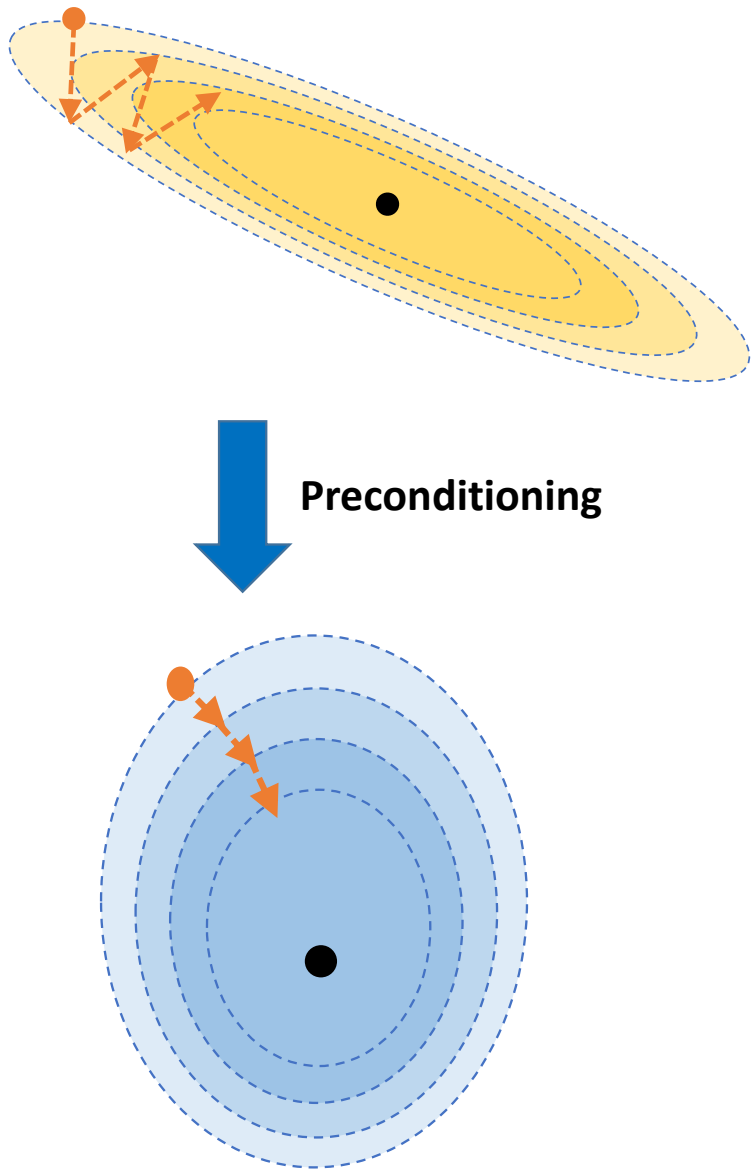


<u>Vanilla GD:</u>

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla f(\boldsymbol{x}_t)$$

☹ **Slows down with ill-conditioning.**

Acceleration via preconditioning



Vanilla GD:

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla f(\boldsymbol{x}_t)$$

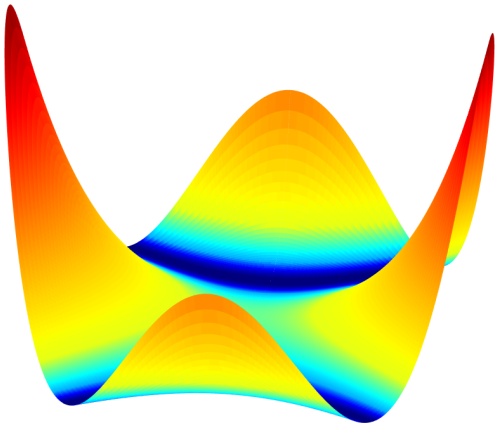
☹ **Slows down with ill-conditioning.**

Preconditioned GD:

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \underbrace{\boldsymbol{H}_t}_{\text{preconditioner}} \nabla f(\boldsymbol{x}_t)$$

☺ **Preconditioning helps!**

Robustness via nonsmooth optimization

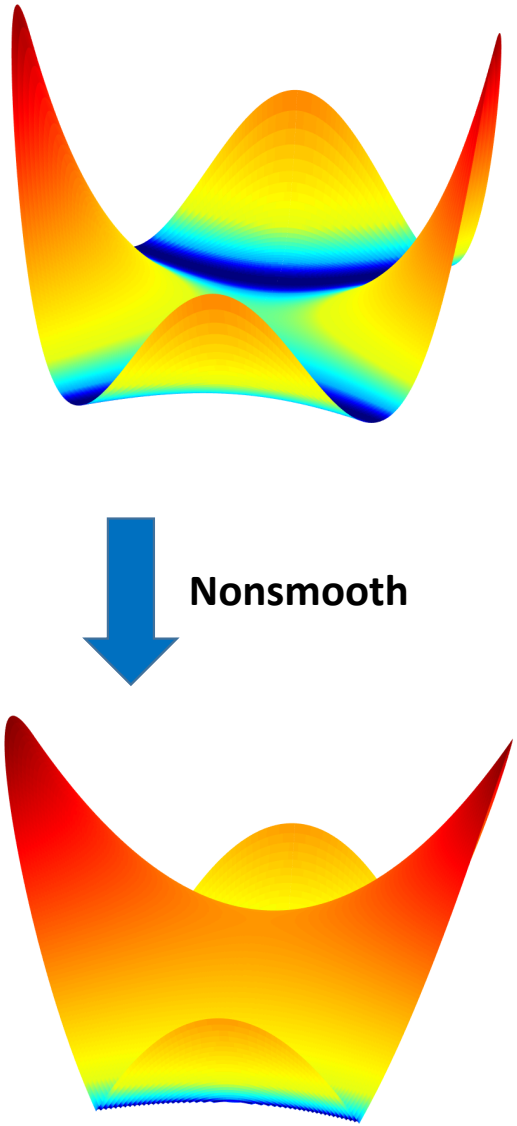


Least squares:

$$f(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_2^2$$

☹ **Sensitive to outliers.**

Robustness via nonsmooth optimization



Least squares:

$$f(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_2^2$$

☹ **Sensitive to outliers.**

Least absolute deviation:

$$f(\boldsymbol{X}, \boldsymbol{Y}) = \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_1$$

☺ **Nonsmoothness helps!**

This talk

Optimization geometry:

When and why does simple gradient descent work well for low-rank matrix estimation?

This talk

Optimization geometry:

When and why does simple gradient descent work well for low-rank matrix estimation?

Acceleration for ill-conditioned matrix estimation:

Can we design provably fast gradient algorithms that are insensitive to the condition number of low-rank matrices?

This talk

Optimization geometry:

When and why does simple gradient descent work well for low-rank matrix estimation?

Acceleration for ill-conditioned matrix estimation:

Can we design provably fast gradient algorithms that are insensitive to the condition number of low-rank matrices?

Robustness to adversarial outliers:

Can we design provably robust gradient algorithms that are oblivious to the presence of outliers?

This talk

Optimization geometry:

When and why does simple gradient descent work well for low-rank matrix estimation?

Acceleration for ill-conditioned matrix estimation:

Can we design provably fast gradient algorithms that are insensitive to the condition number of low-rank matrices?

Robustness to adversarial outliers:

Can we design provably robust gradient algorithms that are oblivious to the presence of outliers?

Generalization to tensors:

Can we generalize to higher-dimensional objects?

A bit preliminaries of optimization

Unconstrained optimization

Consider an unconstrained optimization problem

$$\text{minimize}_{\boldsymbol{x}} \qquad f(\boldsymbol{x})$$

Definition (first-order critical points)

A first-order critical point of f satisfies

$$\nabla f(\boldsymbol{x}) = \mathbf{0}$$

Unconstrained optimization

Consider an unconstrained optimization problem

$$\text{minimize}_{\boldsymbol{x}} \qquad f(\boldsymbol{x})$$

Definition (second-order critical points)

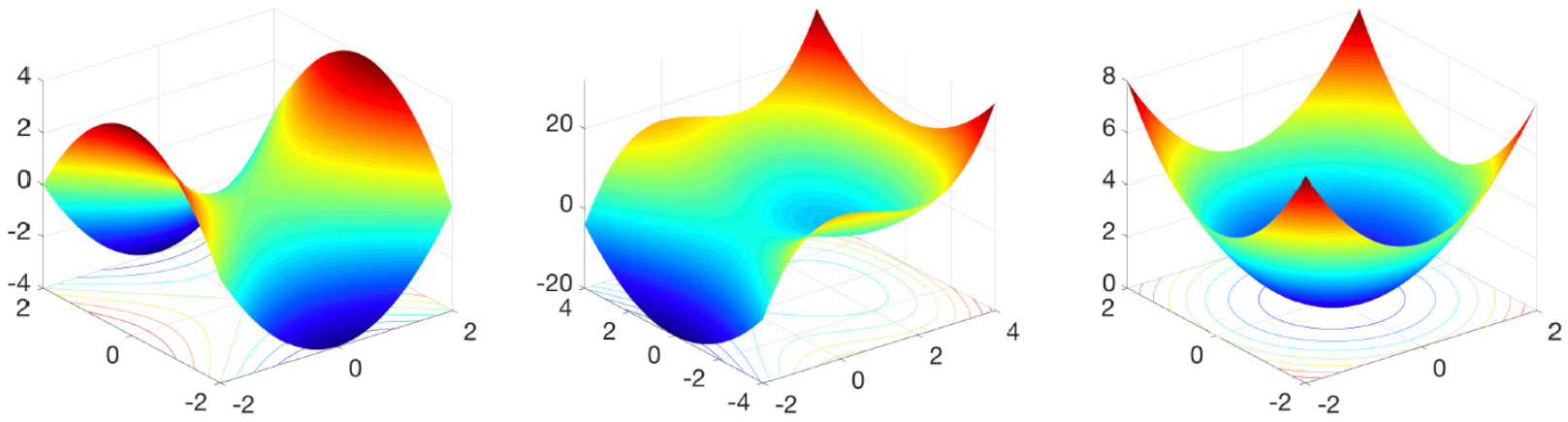
A second-order critical point $\boldsymbol{x}$ satisfies

$$\nabla f(\boldsymbol{x}) = \mathbf{0} \quad \text{and} \quad \nabla^2 f(\boldsymbol{x}) \succeq \mathbf{0}$$

Several types of critical points

For any first-order critical point $\boldsymbol{x}$:

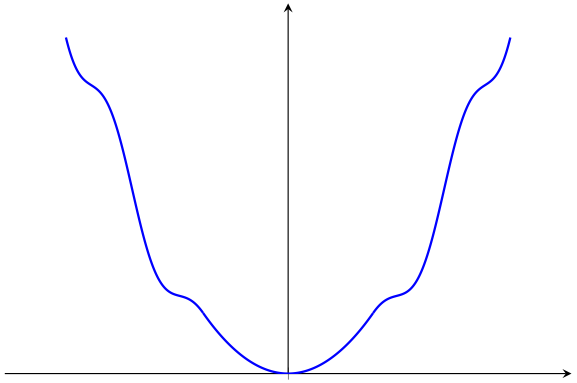
- $\nabla^2 f(\boldsymbol{x}) \prec \mathbf{0}$ $\quad\rightarrow\quad$ local maximum
- $\nabla^2 f(\boldsymbol{x}) \succ \mathbf{0}$ $\quad\rightarrow\quad$ local minimum
- $\lambda_{\min}(\nabla^2 f(\boldsymbol{x})) < 0$ $\quad\rightarrow\quad$ *strict* saddle point



(a) strict saddle (b) local minimum (c) global minimum

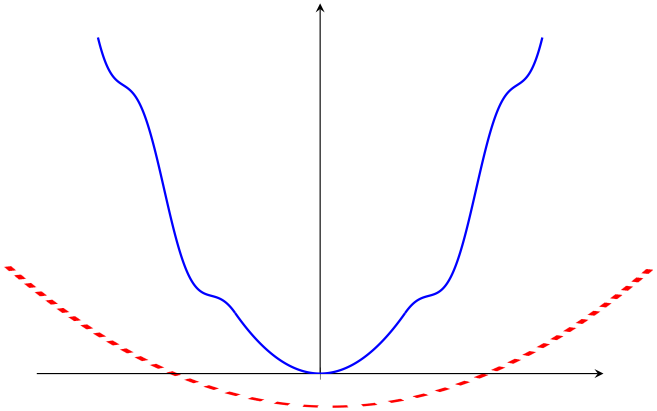
figure credit: Li et al. '16

Gradient descent theory



Two standard conditions that enable geometric convergence of GD

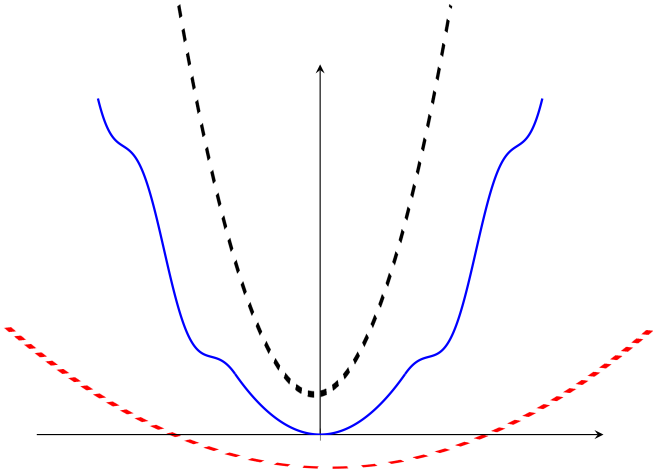
Gradient descent theory



Two standard conditions that enable geometric convergence of GD

- (local) restricted strong convexity (or regularity condition)

Gradient descent theory



Two standard conditions that enable geometric convergence of GD

- (local) restricted strong convexity (or regularity condition)
- (local) smoothness

$$\nabla^2 f(\boldsymbol{x}) \succ \mathbf{0} \quad \text{and} \quad \text{is well-conditioned}$$

Gradient descent theory revisited

f is said to be α-strongly convex and β-smooth if

$$\mathbf{0} \preceq \alpha \boldsymbol{I} \preceq \nabla^2 f(\boldsymbol{x}) \preceq \beta \boldsymbol{I}, \qquad \forall \boldsymbol{x}$$

ℓ_2 **error contraction:** GD ($\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla f(\boldsymbol{x}_t)$) with $\eta = 1/\beta$ obeys

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq \left(1 - \frac{\alpha}{\beta}\right) \|\boldsymbol{x}_t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

Gradient descent theory revisited

f is said to be α-strongly convex and β-smooth if

$$\mathbf{0} \preceq \alpha \boldsymbol{I} \preceq \nabla^2 f(\boldsymbol{x}) \preceq \beta \boldsymbol{I}, \qquad \forall \boldsymbol{x}$$

ℓ_2 error contraction: GD ($\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla f(\boldsymbol{x}_t)$) with $\eta = 1/\beta$ obeys

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq \left(1 - \frac{\alpha}{\beta}\right) \|\boldsymbol{x}_t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

- Condition number β/α determines rate of convergence

Gradient descent theory revisited

f is said to be α-strongly convex and β-smooth if

$$\mathbf{0} \preceq \alpha \boldsymbol{I} \preceq \nabla^2 f(\boldsymbol{x}) \preceq \beta \boldsymbol{I}, \qquad \forall \boldsymbol{x}$$

ℓ_2 error contraction: GD ($\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \nabla f(\boldsymbol{x}_t)$) with $\eta = 1/\beta$ obeys

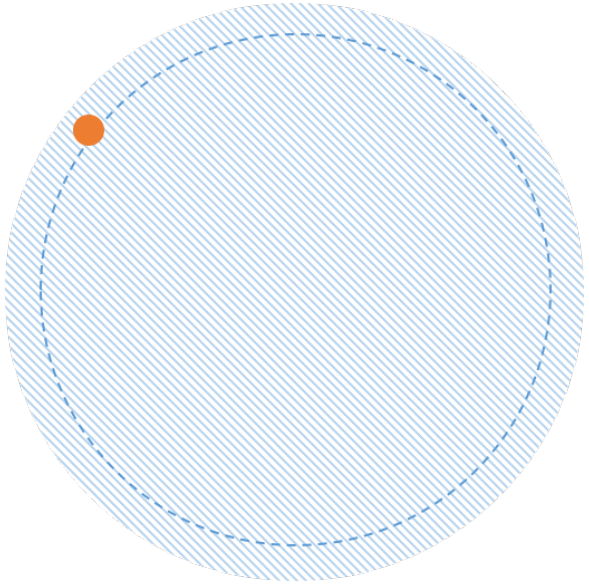
$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq \left(1 - \frac{\alpha}{\beta}\right) \|\boldsymbol{x}_t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

- Condition number β/α determines rate of convergence
- Attains ε-accuracy within $O\big(\frac{\beta}{\alpha} \log \frac{1}{\varepsilon}\big)$ iterations

Gradient descent theory revisited

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq (1 - \alpha/\beta)\, \|\boldsymbol{x}_t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

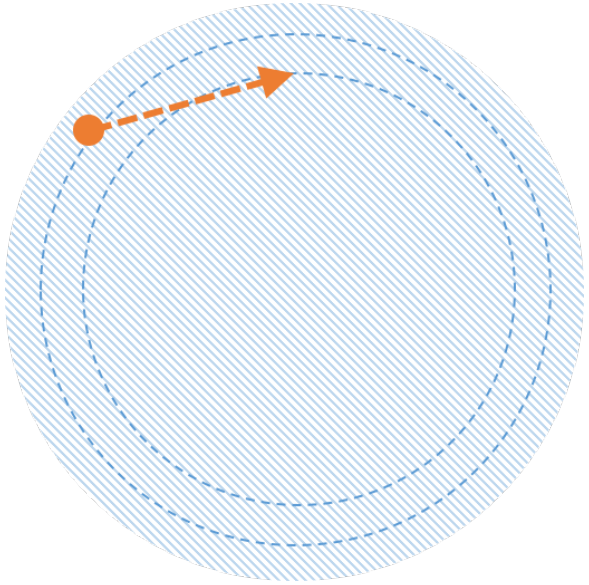
region of local strong convexity + smoothness



Gradient descent theory revisited

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq (1 - \alpha/\beta)\, \|\boldsymbol{x}_t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

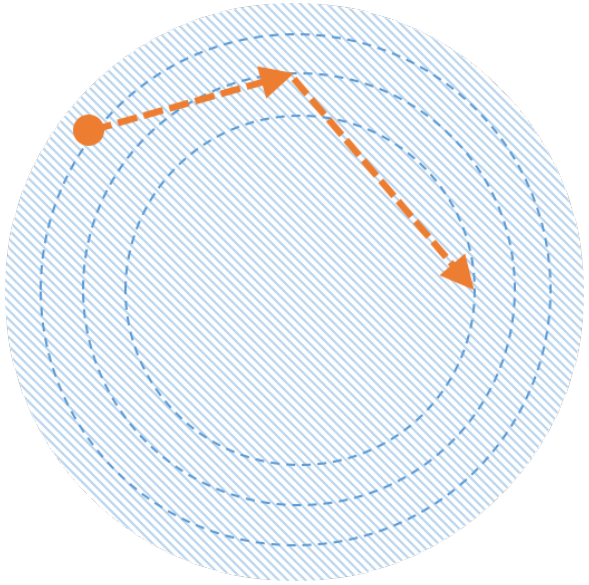
region of local strong convexity + smoothness



Gradient descent theory revisited

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq (1 - \alpha/\beta)\, \|\boldsymbol{x}_t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

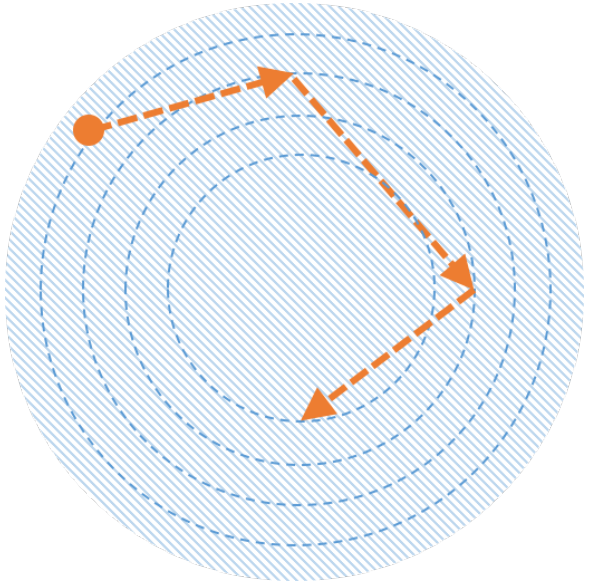
region of local strong convexity + smoothness



Gradient descent theory revisited

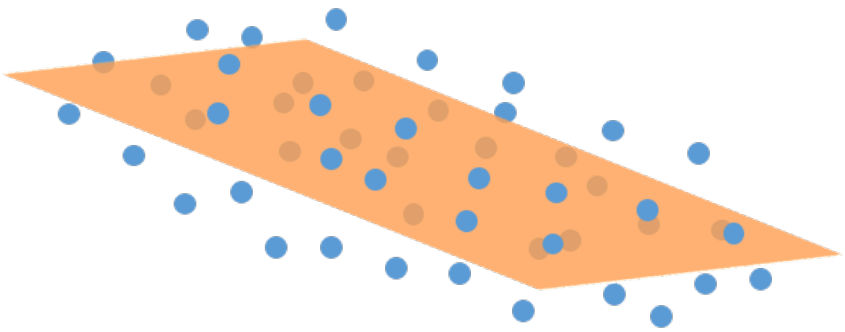
$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}_{\text{opt}}\|_2 \leq (1 - \alpha/\beta)\, \|\boldsymbol{x}_t - \boldsymbol{x}_{\text{opt}}\|_2$$

region of local strong convexity + smoothness



Warm-up: understanding the geometry of PCA

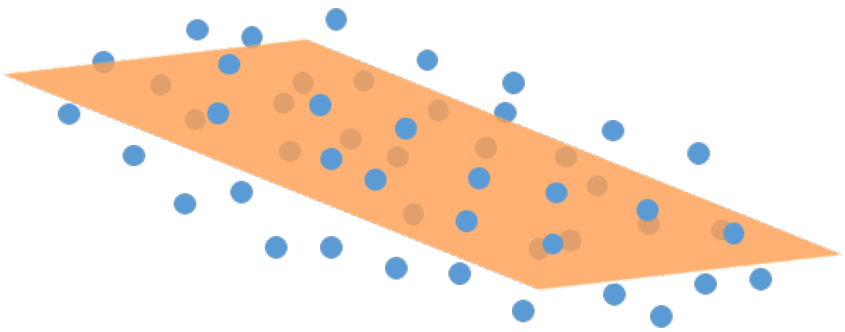
Revisiting PCA



Given $\boldsymbol{M} \succeq \boldsymbol{0} \in \mathbb{R}^{n \times n}$ (not necessarily low-rank), find its best rank-r approximation:

$$\underbrace{\widehat{\boldsymbol{M}} = \text{argmin}_{\boldsymbol{Z}} \, \|\boldsymbol{Z} - \boldsymbol{M}\|_{\mathrm{F}}^2 \quad \text{s.t.} \quad \text{rank}(\boldsymbol{Z}) \leq r}_{\text{nonconvex optimization!}}$$

Revisiting PCA



This problem admits a closed-form solution

- let $\boldsymbol{M} = \sum_{i=1}^{n} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^\top$ be eigen-decomposition of $\boldsymbol{M}$ ($\lambda_1 \geq \cdots \geq \lambda_n$), then

$$\widehat{\boldsymbol{M}} = \sum_{i=1}^{r} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^\top$$

— *nonconvex, but tractable*

An optimization viewpoint

Low-rank factorization: if we factorize $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{X}^\top$ with $\boldsymbol{X} \in \mathbb{R}^{n \times r}$, then it leads to a nonconvex problem:

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \|\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2$$

An optimization viewpoint

Low-rank factorization: if we factorize $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{X}^\top$ with $\boldsymbol{X} \in \mathbb{R}^{n\times r}$, then it leads to a nonconvex problem:

$$\text{minimize}_{\boldsymbol{X}\in\mathbb{R}^{n\times r}} \quad f(\boldsymbol{X}) = \|\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2$$

Theorem (Baldi and Hornik, 1989)

Suppose $\boldsymbol{M}$ has a strict eigen-gap between λ_r and λ_{r+1}, the critical points of $f(\boldsymbol{X})$ can be categorized into

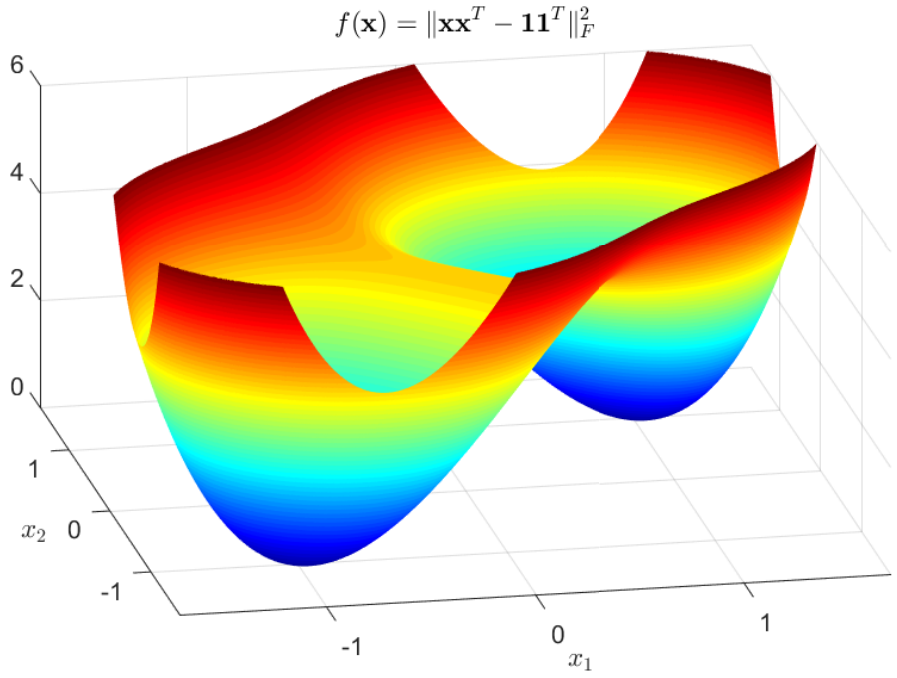
- *global minima;*
- *strict saddle points, from which there exist directions to strictly decrease $f(\boldsymbol{X})$.*

In other words, all local minima are global minima!

Baldi and Hornik. "Neural networks and principal component analysis: Learning from examples without local minima." Neural networks 2.1 (1989): 53-58.

Benign landscape of PCA

For example, for 2-dimensional case $f(\boldsymbol{x}) = \left\| \boldsymbol{x}\boldsymbol{x}^\top - \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \right\|_{\mathrm{F}}^2$



global minima: $\boldsymbol{x} = \pm \begin{bmatrix} 1 \\ 1 \end{bmatrix}$; strict saddles: $\boldsymbol{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, and $\pm \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

— *No "spurious" local minima!*

Local strong convexity and local linear convergence

- The global minimizers: $\boldsymbol{x}_{\text{opt}} = \pm\sqrt{\lambda_1}\boldsymbol{u}_1$
- For all $\boldsymbol{x}$ obeying $\underbrace{\|\boldsymbol{x} - \boldsymbol{x}_{\text{opt}}\|_2 \le \frac{\lambda_1 - \lambda_2}{15\sqrt{\lambda_1}}}_{\text{basin of attraction}}$, one has

$$0.25(\lambda_1 - \lambda_2)\boldsymbol{I}_n \preceq \nabla^2 f(\boldsymbol{x}) \preceq 4.5\lambda_1\boldsymbol{I}_n$$

Local strong convexity and local linear convergence

- The global minimizers: $\boldsymbol{x}_{\text{opt}} = \pm\sqrt{\lambda_1}\boldsymbol{u}_1$
- For all $\boldsymbol{x}$ obeying $\underbrace{\|\boldsymbol{x} - \boldsymbol{x}_{\text{opt}}\|_2 \le \frac{\lambda_1 - \lambda_2}{15\sqrt{\lambda_1}}}_{\text{basin of attraction}}$, one has

$$0.25(\lambda_1 - \lambda_2)\boldsymbol{I}_n \preceq \nabla^2 f(\boldsymbol{x}) \preceq 4.5\lambda_1 \boldsymbol{I}_n$$

ℓ_2 **error contraction:** The GD iterates obey

$$\left\|\boldsymbol{x}_t - \sqrt{\lambda_1}\boldsymbol{u}_1\right\|_2 \le \left(1 - \frac{\lambda_1 - \lambda_2}{18\lambda_1}\right)^t \left\|\boldsymbol{x}_0 - \sqrt{\lambda_1}\boldsymbol{u}_1\right\|_2, \ t \ge 0,$$

as long as $\|\boldsymbol{x}_0 - \sqrt{\lambda_1}\boldsymbol{u}_1\|_2 \le \frac{\lambda_1 - \lambda_2}{15\sqrt{\lambda_1}}$

Extension to the low-rank case

$$f(\boldsymbol{X}) := \frac{1}{4}\|\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2, \qquad \boldsymbol{X} \in \mathbb{R}^{n\times r}$$

Cannot be uniquely determined $\boldsymbol{X}$ up to orthogonal transform.

Extension to the low-rank case

$$f(\boldsymbol{X}) := \frac{1}{4}\|\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2, \qquad \boldsymbol{X} \in \mathbb{R}^{n\times r}$$

Cannot be uniquely determined $\boldsymbol{X}$ up to orthogonal transform.

- A modified distance metric:

$$\mathrm{dist}^2(\boldsymbol{X}, \boldsymbol{X}_\star) = \min_{\boldsymbol{H}\in\mathcal{O}^{r\times r}} \|\boldsymbol{X}\boldsymbol{H} - \boldsymbol{X}_\star\|_{\mathrm{F}}^2.$$

- Optimal alignment matrix (the Procruste problem):

$$\boldsymbol{H}_{\boldsymbol{X}} := \operatorname*{argmin}_{\boldsymbol{H}\in\mathcal{O}^{r\times r}} \|\boldsymbol{X}\boldsymbol{H} - \boldsymbol{X}_\star\|_{\mathrm{F}}^2.$$

Restricted strong convexity

$$f(\boldsymbol{X}) := \frac{1}{4}\|\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2, \qquad \boldsymbol{X} \in \mathbb{R}^{n\times r}$$

f satisfies α-**restricted strong convexity** and β-smoothness:

$$\mathrm{vec}(\boldsymbol{V})^\top \nabla^2 f(\boldsymbol{X})\mathrm{vec}(\boldsymbol{V}) \geq \alpha\|\boldsymbol{V}\|_{\mathrm{F}}^2, \quad \boldsymbol{V} := \boldsymbol{X}\boldsymbol{H}_{\boldsymbol{X}} - \boldsymbol{X}_\star$$

where $\beta \asymp \lambda_1$ and $\alpha \asymp \lambda_r$.

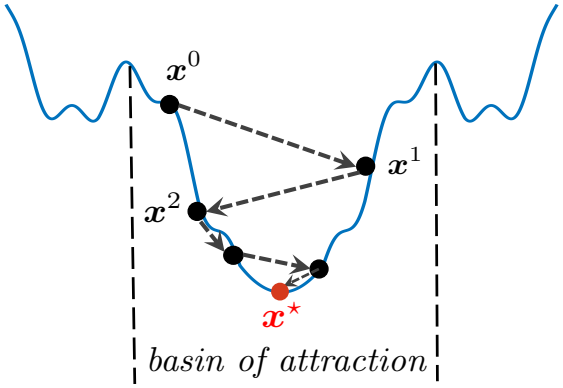
ℓ_2 **error contraction:** The GD iterates obey

$$\mathrm{dist}^2(\boldsymbol{X}_t, \boldsymbol{X}_\star) \leq \left(1 - \frac{c}{\kappa}\right)^t \mathrm{dist}^2(\boldsymbol{X}_0, \boldsymbol{X}_\star), \qquad t \geq 0,$$

as long as $\mathrm{dist}^2(\boldsymbol{X}_0, \boldsymbol{X}_\star) \lesssim \lambda_1$. Here, $\kappa := \lambda_1/\lambda_r$.

Two vignettes

Two-stage approach:



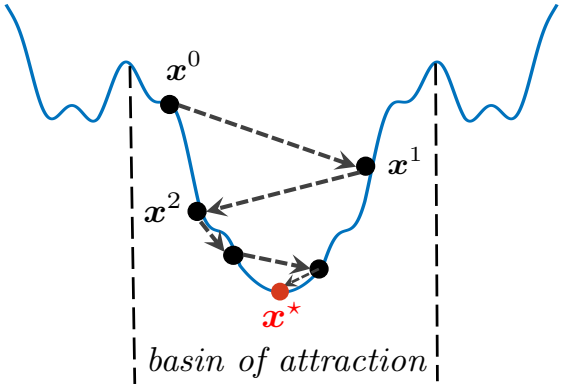
smart initialization

$+$

local refinement

Two vignettes

Two-stage approach:

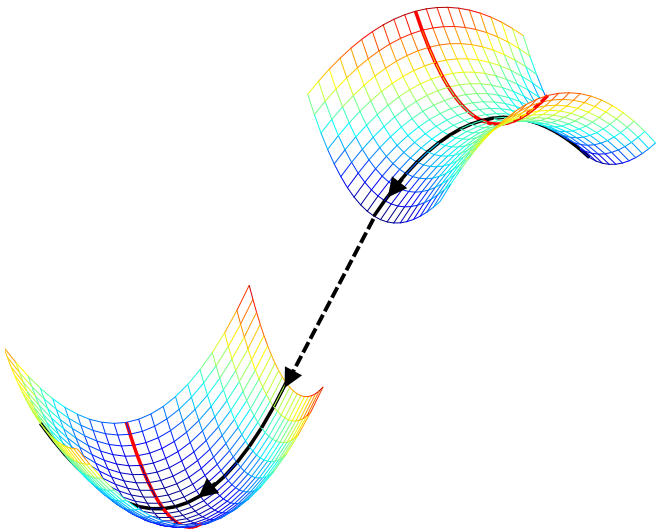


smart initialization

$+$

local refinement

Global landscape:



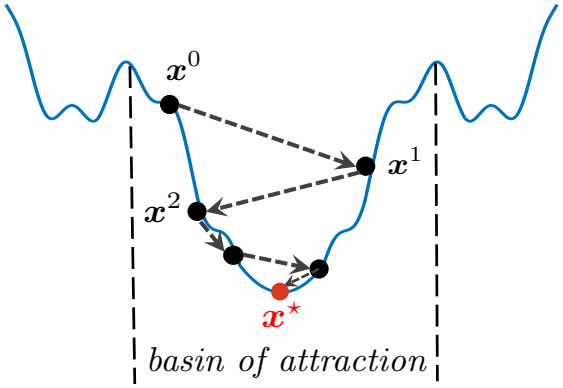
benign landscape

$+$

saddle-point escaping

Two vignettes

Two-stage approach:

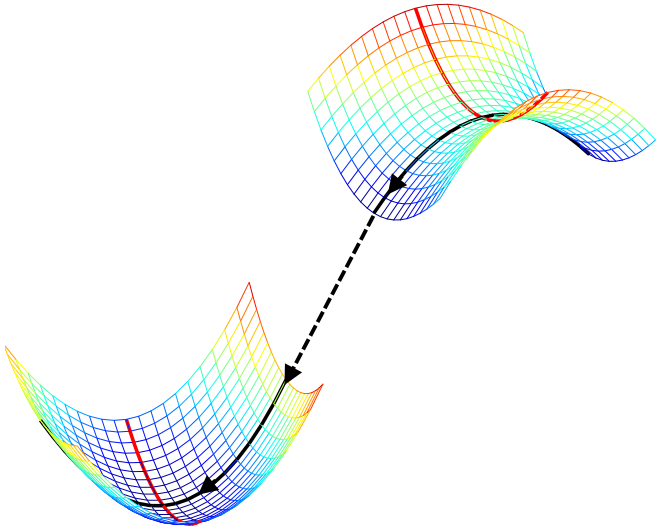


smart initialization

$+$

local refinement

Global landscape:



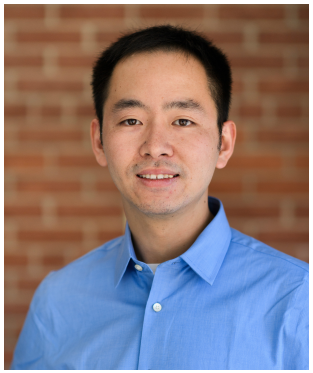
benign landscape

$+$

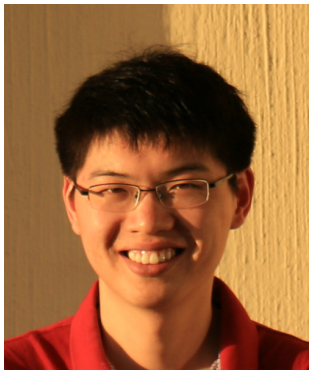
saddle-point escaping

This tutorial will mostly focus on the two-stage approach.

Geometry and implicit regularization in nonconvex low-rank matrix estimation



Yuxin Chen
Princeton



Cong Ma
Chicago

Kaizheng Wang
Columbia

Low-rank matrix completion: dealing with missing data

$$\begin{bmatrix} \checkmark & ? & ? & ? & \checkmark & ? \\ ? & ? & \checkmark & \checkmark & ? & ? \\ \checkmark & ? & ? & \checkmark & ? & ? \\ ? & ? & \checkmark & ? & ? & \checkmark \\ \checkmark & ? & ? & ? & ? & ? \\ ? & \checkmark & ? & ? & \checkmark & ? \\ ? & ? & \checkmark & \checkmark & ? & ? \end{bmatrix}$$

Given partial samples of a *low-rank* matrix $\boldsymbol{M} = \boldsymbol{X}_\star \boldsymbol{X}_\star^\top \in \mathbb{R}^{n \times n}$ in an index set Ω, fill in missing entries.

A natural least-squares formulation

$$\text{given:} \quad \mathcal{P}_\Omega(\boldsymbol{M})$$

$$\Downarrow$$

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \left\| \mathcal{P}_\Omega(\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}) \right\|_{\mathrm{F}}^2$$

A natural least-squares formulation

$$\text{given:} \quad \mathcal{P}_\Omega(\boldsymbol{M})$$

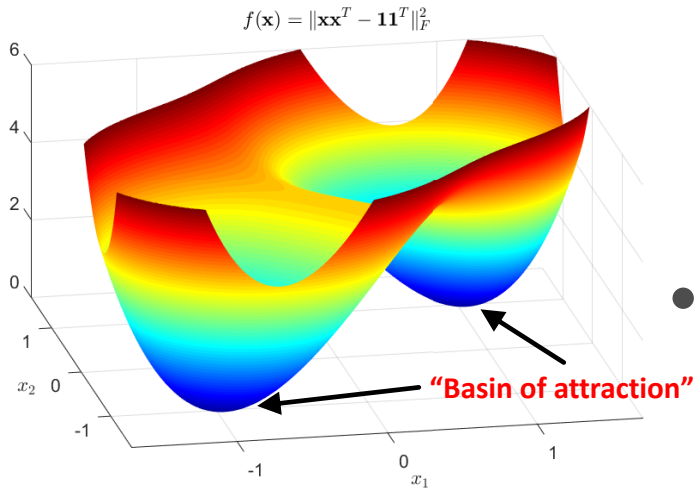
$$\Downarrow$$

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \left\| \mathcal{P}_\Omega(\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M}) \right\|_{\mathrm{F}}^2$$

- **Bernoulli sampling:** *Assume every entry is observed i.i.d. with* $0 < p \leq 1$*:*

$$\mathbb{E}[f(\boldsymbol{X})] = p \left\| \boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M} \right\|_{\mathrm{F}}^2 .$$

Two-stage approach



- **Spectral initialization:** find an initial point in the "basin of attraction".

$$\boldsymbol{X}_0 = \mathsf{SVD}_r(\mathcal{P}_\Omega(\boldsymbol{M}))$$

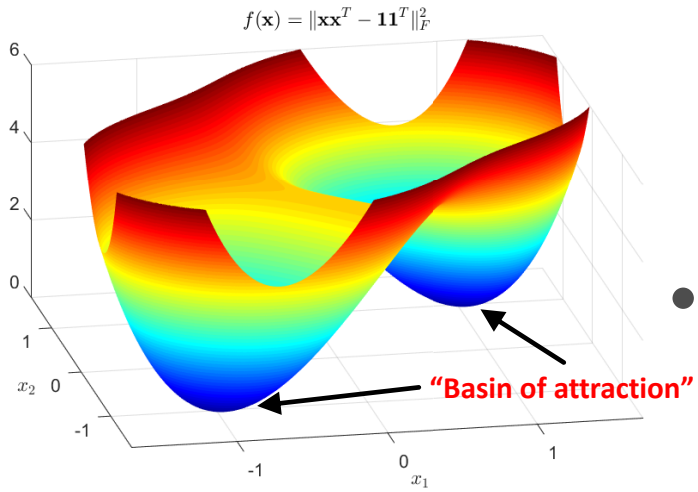
- **Gradient iterations:**

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla f(\boldsymbol{X}_t)$$

for $t = 0, 1, \ldots$

"Spectral methods for data science: A statistical perspective", Y. Chen, Y. Chi, J. Fan, C. Ma, FnT ML, 2021.

Two-stage approach



- **Spectral initialization:** find an initial point in the "basin of attraction".

$$\boldsymbol{X}_0 = \mathsf{SVD}_r(\mathcal{P}_\Omega(\boldsymbol{M}))$$

- **Gradient iterations:**

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla f(\boldsymbol{X}_t)$$

for $t = 0, 1, \ldots$

Question: Does vanilla GD still work with partial observations?

"Spectral methods for data science: A statistical perspective", Y. Chen, Y. Chi, J. Fan, C. Ma, FnT ML, 2021.

Incoherence

Which is easier to complete?

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}} \quad \text{vs.} \quad \underbrace{\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & & \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}}$$

Incoherence

Which is easier to complete?

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}}_{\text{hard}} \quad \text{vs.} \quad \underbrace{\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & & \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}}_{\text{easy}}$$

Definition (Incoherence for matrix completion)

A rank-r positive-semidefinite matrix $\boldsymbol{M}$ with eigendecomposition $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^\top$ is said to be μ-incoherent if

$$\|\boldsymbol{U}\|_{2,\infty} \leq \sqrt{\frac{\mu}{n}}\,\|\boldsymbol{U}\|_{\mathrm{F}} = \sqrt{\frac{\mu r}{n}}.$$

Incoherence

Which is easier to complete?

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}}_{\text{hard } \mu=n} \quad \text{vs.} \quad \underbrace{\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & & \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}}_{\text{easy } \mu=1}$$

Definition (Incoherence for matrix completion)

A rank-r positive-semidefinite matrix $\boldsymbol{M}$ with eigendecomposition $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^\top$ is said to be μ-incoherent if

$$\|\boldsymbol{U}\|_{2,\infty} \leq \sqrt{\frac{\mu}{n}}\,\|\boldsymbol{U}\|_{\mathrm{F}} = \sqrt{\frac{\mu r}{n}}.$$

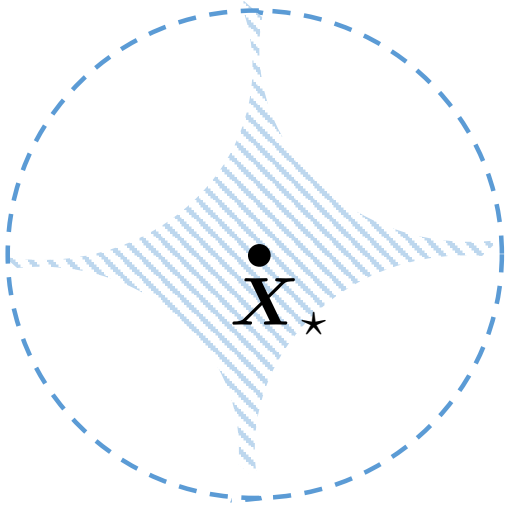
Which region has benign geometry?

Finite-sample level ($p \asymp \frac{\text{polylog} n}{n}$) **:**

$f(\boldsymbol{X})$ restricted strongly convex and smooth

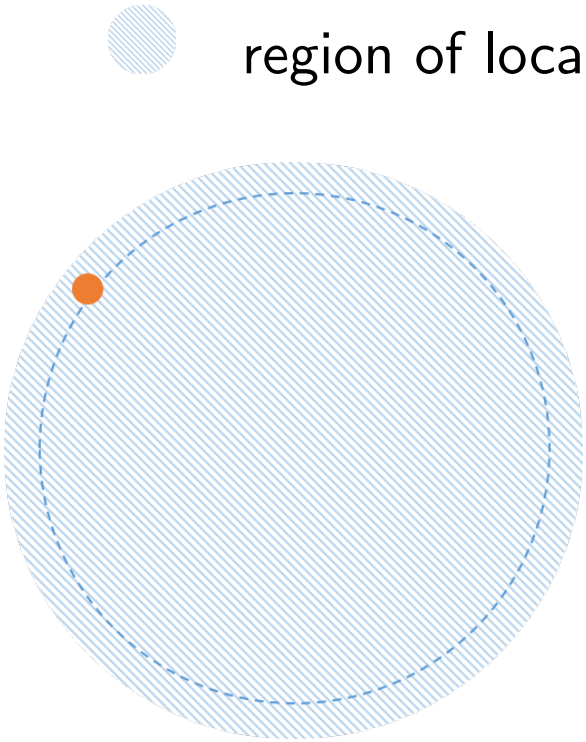
along descent direction $\boldsymbol{V}$ only when $\boldsymbol{X}$ is incoherent:

$$\|\boldsymbol{X}\boldsymbol{H}_{\boldsymbol{X}} - \boldsymbol{X}_\star\|_{2,\infty} \ll \|\boldsymbol{X}_\star\|_{2,\infty}$$



region of local strong convexity + smoothness

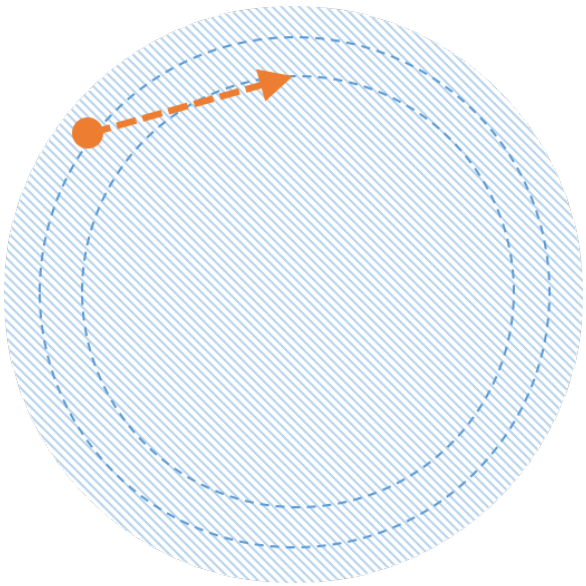
Vanilla gradient descent is at risk



GD on the pop. loss

Vanilla gradient descent is at risk

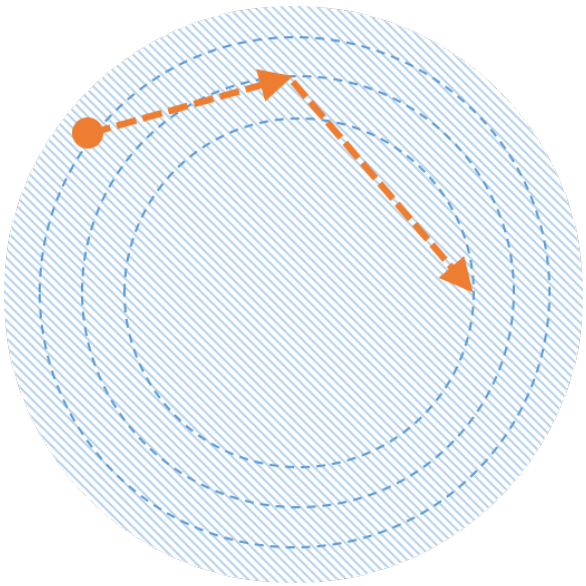
region of local strong convexity + smoothness



GD on the pop. loss

Vanilla gradient descent is at risk

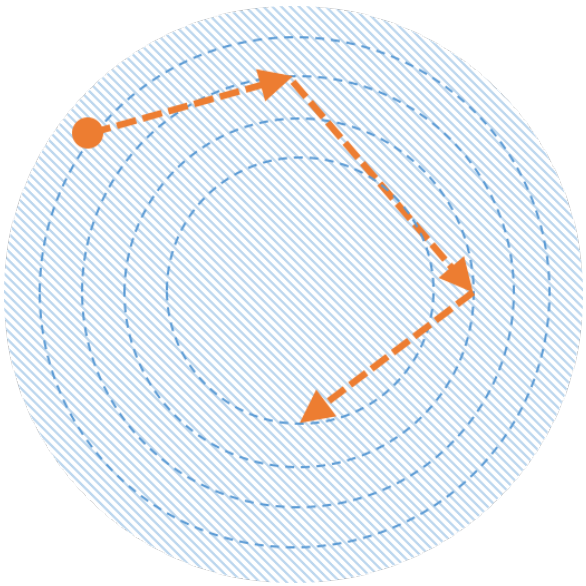
region of local strong convexity + smoothness



GD on the pop. loss

Vanilla gradient descent is at risk

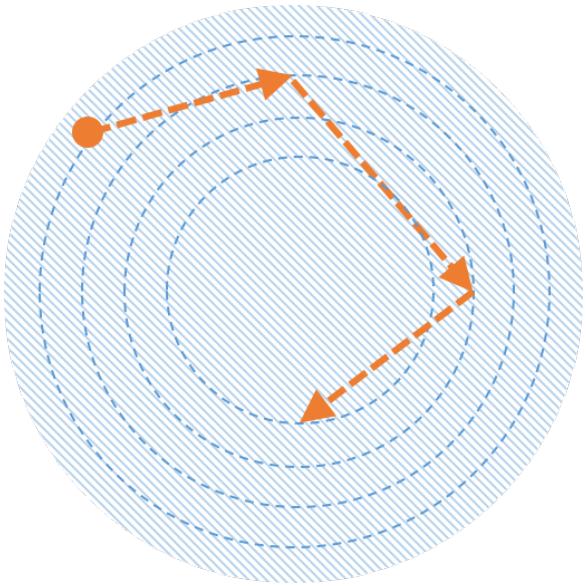
region of local strong convexity + smoothness



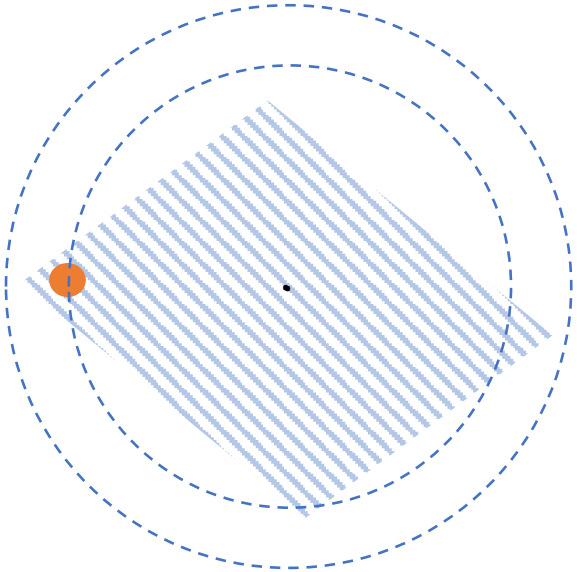
GD on the pop. loss

Vanilla gradient descent is at risk

region of local strong convexity + smoothness



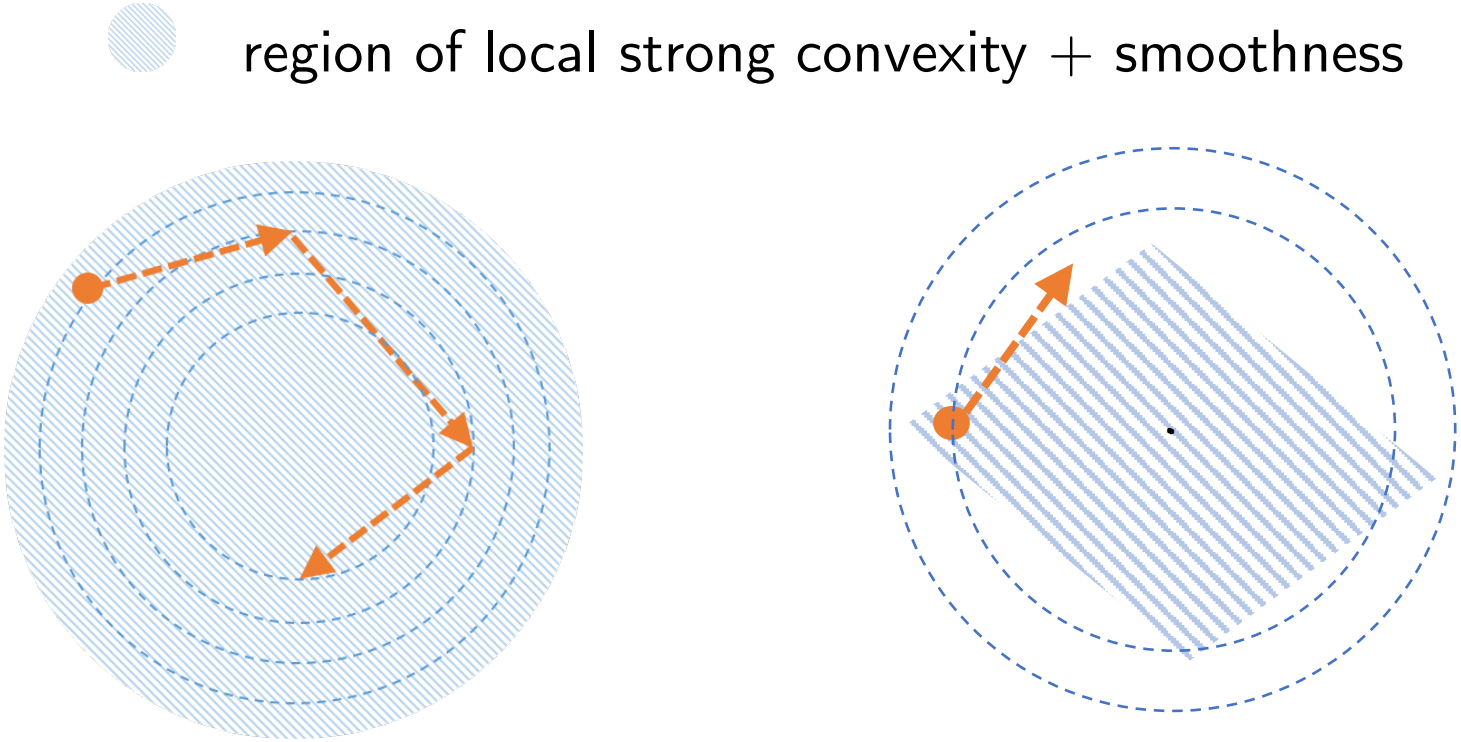
GD on the pop. loss



GD on the emp. loss

- Generic optimization theory only ensures that iterates remain in ℓ_2 ball but not incoherence region

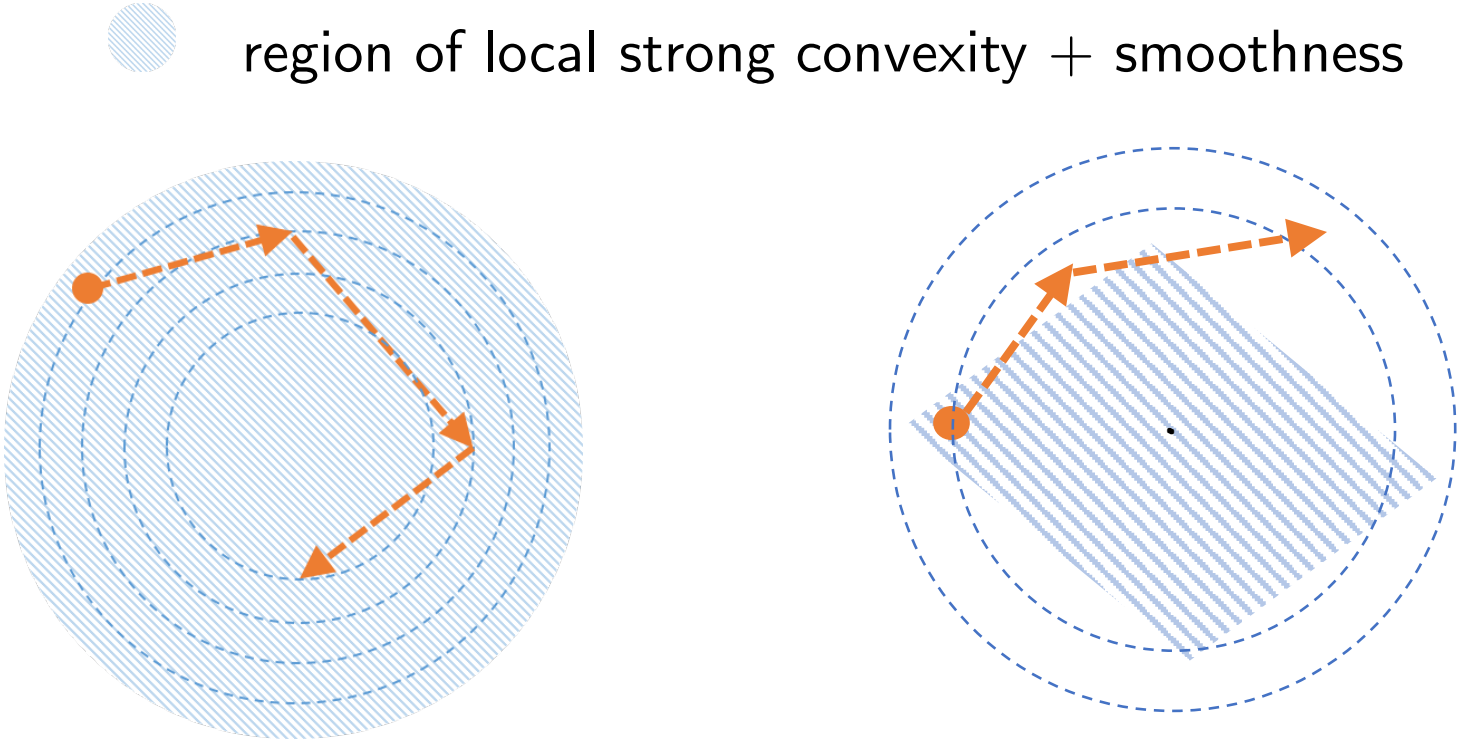
Vanilla gradient descent is at risk



GD on the pop. loss *GD on the emp. loss*

- Generic optimization theory only ensures that iterates remain in ℓ_2 ball but not incoherence region

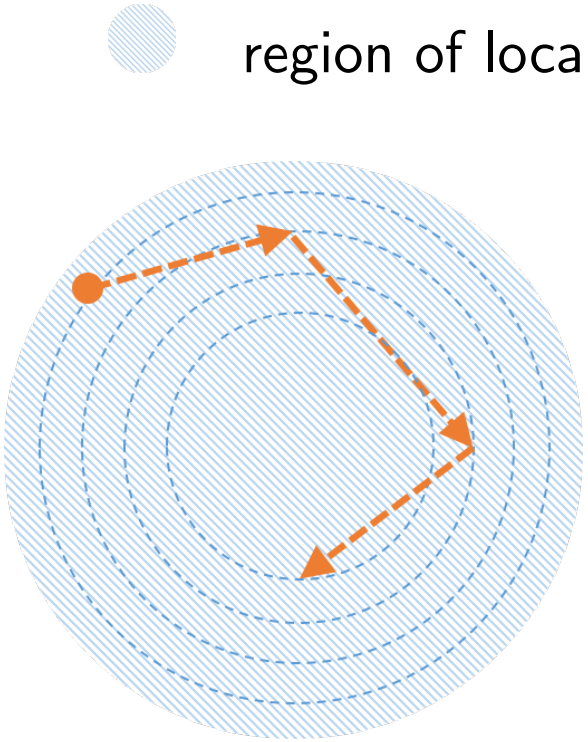
Vanilla gradient descent is at risk

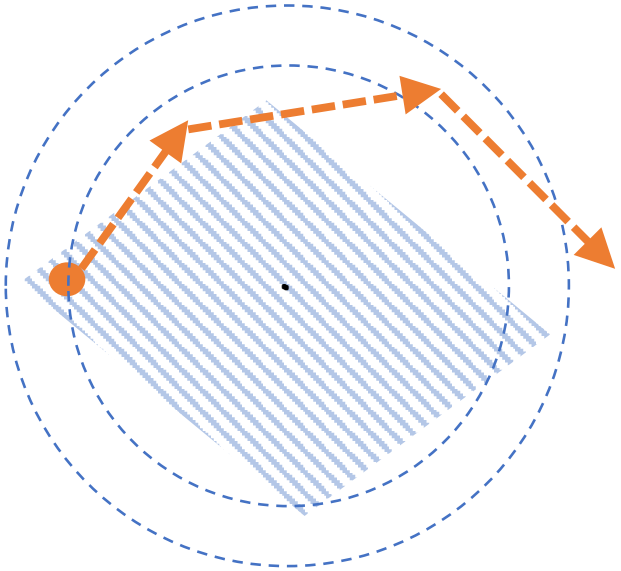


GD on the pop. loss *GD on the emp. loss*

- Generic optimization theory only ensures that iterates remain in ℓ_2 ball but not incoherence region

Vanilla gradient descent is at risk

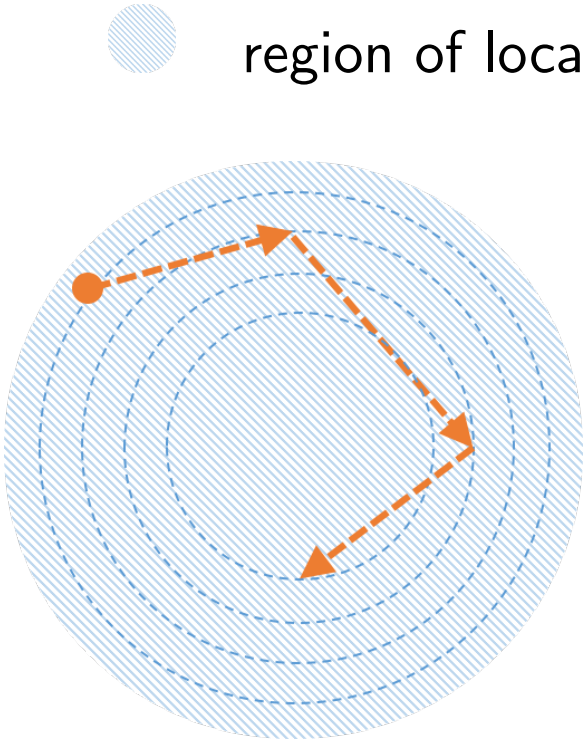




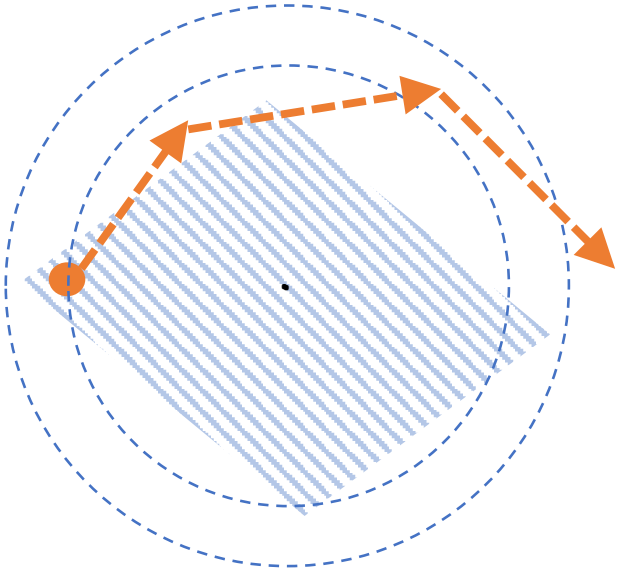
GD on the pop. loss *GD on the emp. loss*

- Generic optimization theory only ensures that iterates remain in ℓ_2 ball but not incoherence region

Vanilla gradient descent is at risk



region of local strong convexity + smoothness



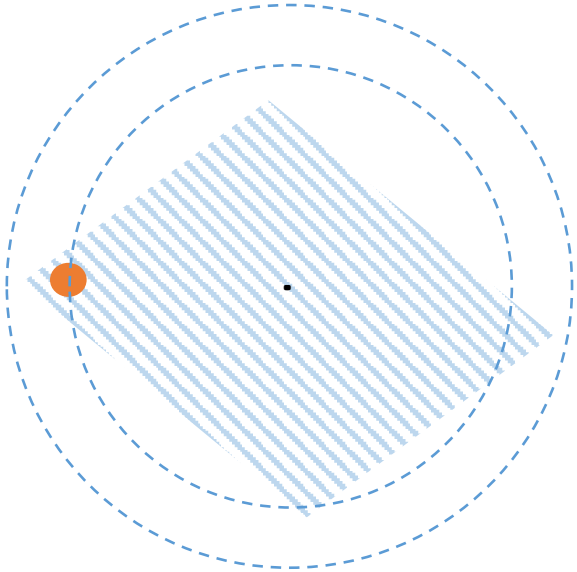
GD on the pop. loss

GD on the emp. loss

- Generic optimization theory only ensures that iterates remain in ℓ_2 ball but not incoherence region
- *Existing algorithms enforce regularization, or apply sample splitting to promote incoherence*

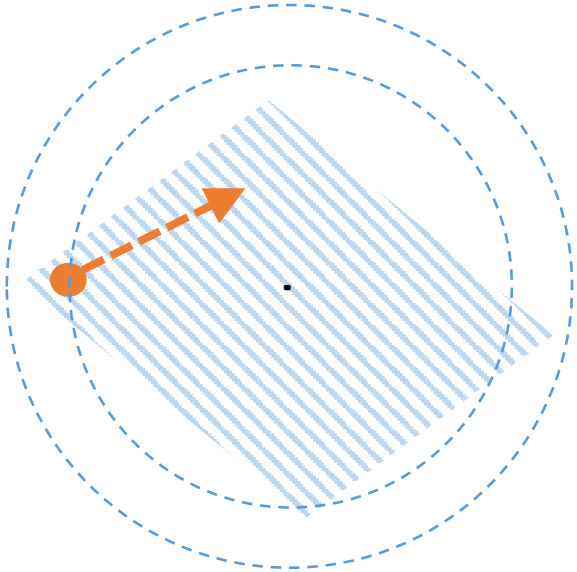
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness



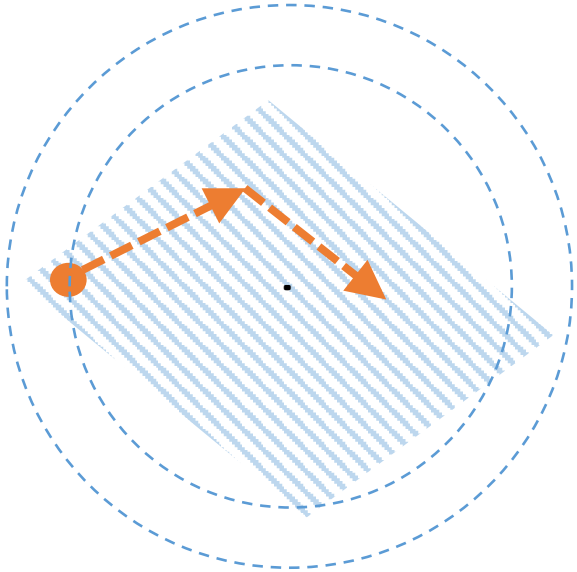
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness



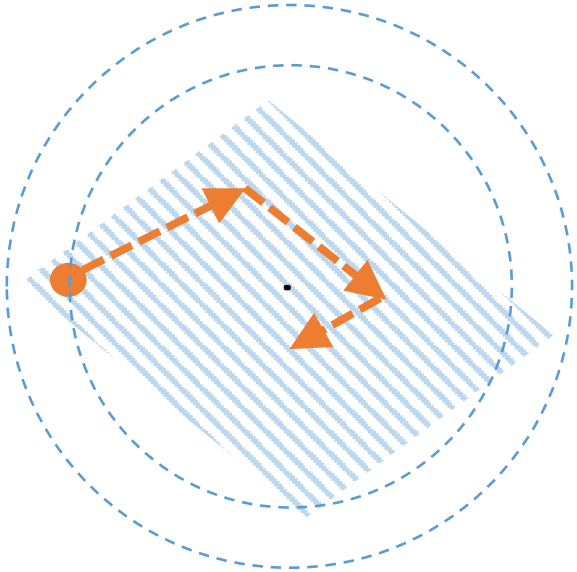
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness



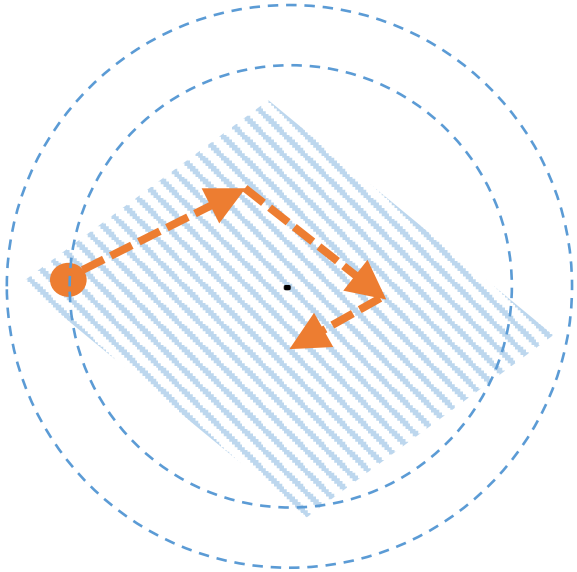
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness



Our findings: GD is implicitly regularized

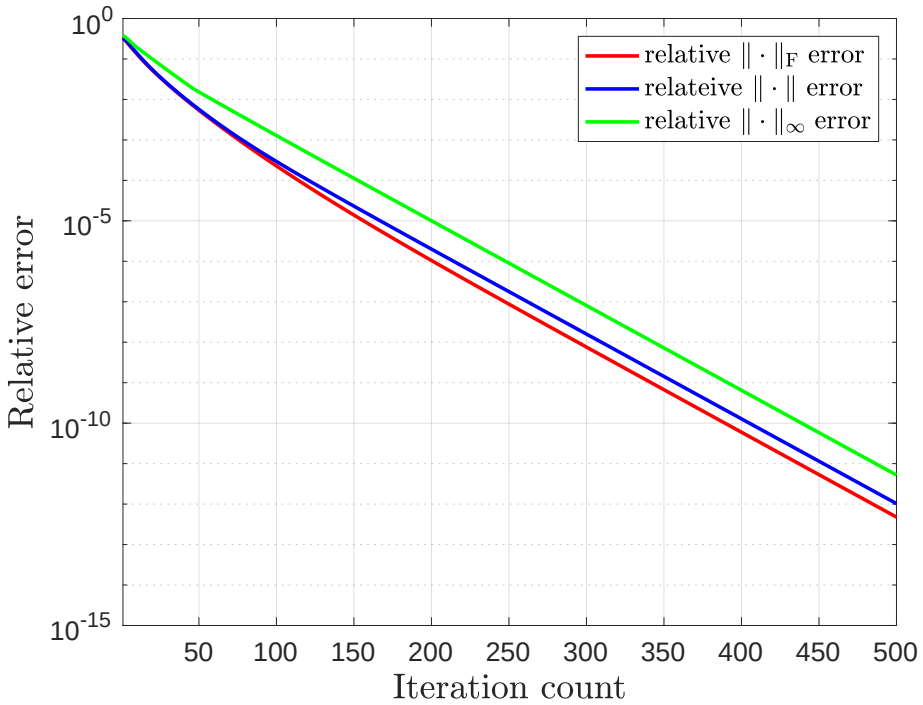
region of local strong convexity + smoothness



GD implicitly forces iterates to remain incoherent even without regularization

Matrix completion via vanilla GD

$$\text{minimize}_{\boldsymbol{X}\in\mathbb{R}^{n\times r}} \quad f(\boldsymbol{X}) = \left\|\mathcal{P}_\Omega(\boldsymbol{X}\boldsymbol{X}^\top - \boldsymbol{M})\right\|_{\mathrm{F}}^2$$



Vanilla GD converges fast without regularization!

Theoretical guarantees - noise-free case

Theorem (Ma, Wang, Chi, Chen, FoCM 2020)

Suppose $\boldsymbol{M} = \boldsymbol{X}_\star \boldsymbol{X}_\star^\top$ *is rank-*r*,* μ*-incoherent and has a condition number* $\kappa = \sigma_{\max}(\boldsymbol{M})/\sigma_{\min}(\boldsymbol{M})$*. Vanilla GD (with spectral initialization) achieves*

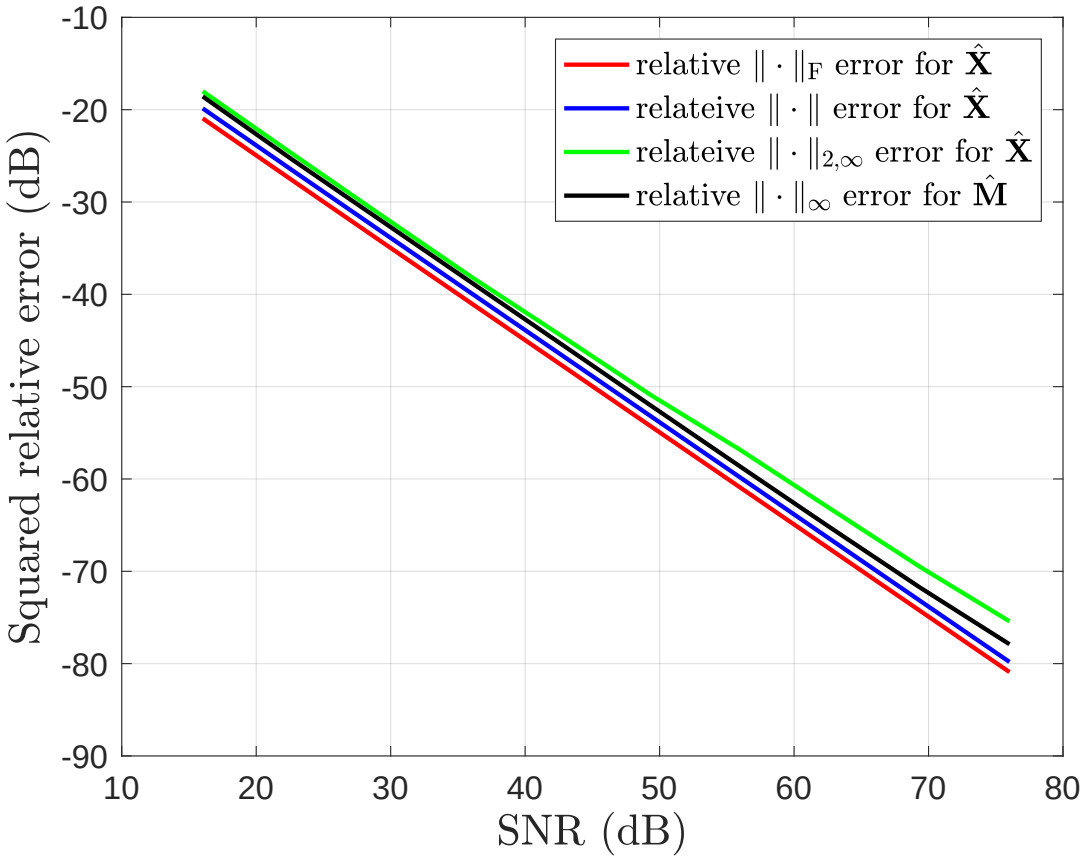
$$\|\boldsymbol{X}_t \boldsymbol{X}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \leq \varepsilon \cdot \sigma_{\min}(\boldsymbol{M})$$

- **Computational:** *within* $O\big(\kappa \log \frac{1}{\varepsilon}\big)$ *iterations;*
- **Statistical:** *as long as the sample complexity satisfies*

$$n^2 p \gtrsim n r^3 \mathsf{poly}(\mu, \kappa, \log n).$$

First convergence guarantee of vanilla GD for matrix completion

Noisy matrix completion via vanilla GD



Near-optimal entrywise error control:

$$\left\| \boldsymbol{X}_t \boldsymbol{X}_t^\top - \boldsymbol{M} \right\|_\infty \lesssim \left(\rho^t \mu r \sqrt{\frac{\log n}{np}} + \frac{\sigma}{\sigma_{\min}} \sqrt{\frac{n \log n}{p}} \right) \|\boldsymbol{M}\|_\infty$$

The phenomenon is quite general

	Prior theory		Our theory	
	sample complexity	iteration complexity	sample complexity	iteration complexity
Phase retrieval	$n \log n$	$n \log\left(\frac{1}{\varepsilon}\right)$	$n \log n$	$\log n \log\left(\frac{1}{\varepsilon}\right)$
Quadratic sensing	$nr^6 \log^2 n$	$n^4 r^2 \log\left(\frac{1}{\varepsilon}\right)$	$nr^4 \log n$	$r^2 \log\left(\frac{1}{\varepsilon}\right)$
Matrix completion	n/a	n/a	$nr^3 \text{poly} \log n$	$\log\left(\frac{1}{\varepsilon}\right)$
Blind deconvolution	n/a	n/a	$K \text{poly} \log m$	$\log\left(\frac{1}{\varepsilon}\right)$

Huge computational savings!

An aside: minimax stability of nuclear norm minimization

convex

nonconvex

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{n\times n}} \sum_{(i,j)\in\Omega} \left(Z_{i,j} - M_{i,j}\right)^2 + \lambda\|\boldsymbol{Z}\|_*$$

$$\min_{\boldsymbol{X},\boldsymbol{Y}\in\mathbb{R}^{n\times r}} \sum_{(i,j)\in\Omega} \left[\left(\boldsymbol{X}\boldsymbol{Y}^\top\right)_{i,j} - M_{i,j}\right]^2 + \tfrac{\lambda}{2}\|\boldsymbol{X}\|_{\mathrm{F}}^2 + \tfrac{\lambda}{2}\|\boldsymbol{Y}\|_{\mathrm{F}}^2$$

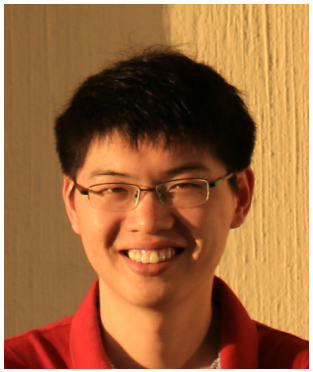
Theorem (Chen, Chi, Fan, Ma, Yan '19)

With high prob., any minimizer $\widehat{\boldsymbol{M}}_{\mathsf{cvx}}$ of convex program is nearly rank-r and is minimax near-optimal:

$$\left\|\widehat{\boldsymbol{M}}_{\mathsf{cvx}} - \boldsymbol{M}\right\|_{\mathrm{F}} \lesssim \sigma\sqrt{\tfrac{n}{p}}, \quad \left\|\widehat{\boldsymbol{M}}_{\mathsf{cvx}} - \boldsymbol{M}\right\|_{\infty} \lesssim \sigma\sqrt{\frac{n\log n}{p}} \cdot \frac{1}{n}$$

Noisy Matrix Completion: Understanding Statistical Guarantees for Convex Relaxation via Nonconvex Optimization, SIAM Journal on Optimization.

Accelerating ill-conditioned matrix estimation

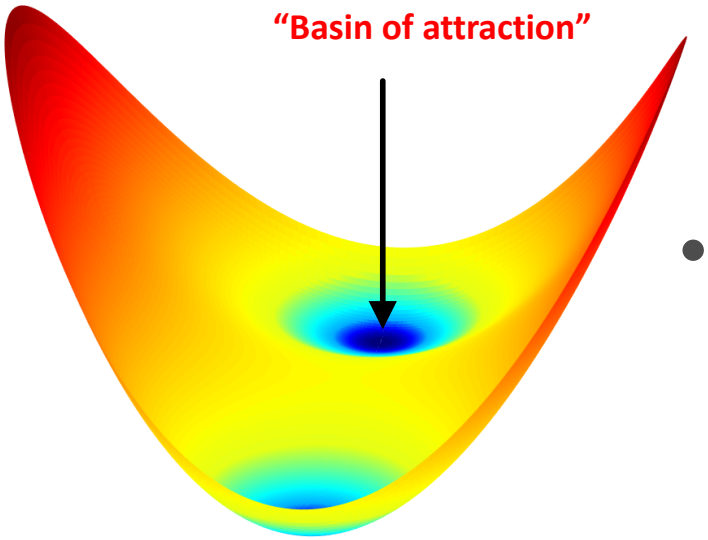


Tian Tong
CMU

Cong Ma
Chicago

The asymmetric case: GD with balancing regularization

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad f_{\text{reg}}(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\left\|\boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top)\right\|_2^2 + \frac{1}{8}\left\|\boldsymbol{X}^\top\boldsymbol{X} - \boldsymbol{Y}^\top\boldsymbol{Y}\right\|_{\text{F}}^2$$



- **Spectral initialization:** find an initial point in the "basin of attraction".

$$(\boldsymbol{X}_0, \boldsymbol{Y}_0) \leftarrow \mathsf{SVD}_r(\mathcal{A}^*(\boldsymbol{y}))$$

- **Gradient iterations:**

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta\,\nabla_{\boldsymbol{X}} f_{\text{reg}}(\boldsymbol{X}_t, \boldsymbol{Y}_t)$$
$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta\,\nabla_{\boldsymbol{Y}} f_{\text{reg}}(\boldsymbol{X}_t, \boldsymbol{Y}_t)$$

for $t = 0, 1, \ldots$

GD for asymmetric low-rank matrix sensing

Theorem (Tu et al., ICML 2016)

Suppose $\boldsymbol{M} = \boldsymbol{X}_\star \boldsymbol{Y}_\star^\top$ *is rank-*r *and has a condition number* $\kappa = \sigma_{\max}(\boldsymbol{M})/\sigma_{\min}(\boldsymbol{M})$. *For low-rank matrix sensing with i.i.d. Gaussian design, vanilla GD (with spectral initialization) achieves*

$$\|\boldsymbol{X}_t \boldsymbol{Y}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \leq \varepsilon \cdot \sigma_{\min}(\boldsymbol{M})$$

- **Computational:** *within* $O\big(\kappa \log \frac{1}{\varepsilon}\big)$ *iterations;*
- **Statistical:** *as long as the sample complexity satisfies*

$$m \gtrsim (n_1 + n_2) r^2 \kappa^2.$$

GD for asymmetric low-rank matrix sensing

Theorem (Tu et al., ICML 2016)

Suppose $\boldsymbol{M} = \boldsymbol{X}_\star \boldsymbol{Y}_\star^\top$ *is rank-*r *and has a condition number* $\kappa = \sigma_{\max}(\boldsymbol{M})/\sigma_{\min}(\boldsymbol{M})$. *For low-rank matrix sensing with i.i.d. Gaussian design, vanilla GD (with spectral initialization) achieves*

$$\|\boldsymbol{X}_t \boldsymbol{Y}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \leq \varepsilon \cdot \sigma_{\min}(\boldsymbol{M})$$

- **Computational:** *within* $O\big(\kappa \log \frac{1}{\varepsilon}\big)$ *iterations;*
- **Statistical:** *as long as the sample complexity satisfies*

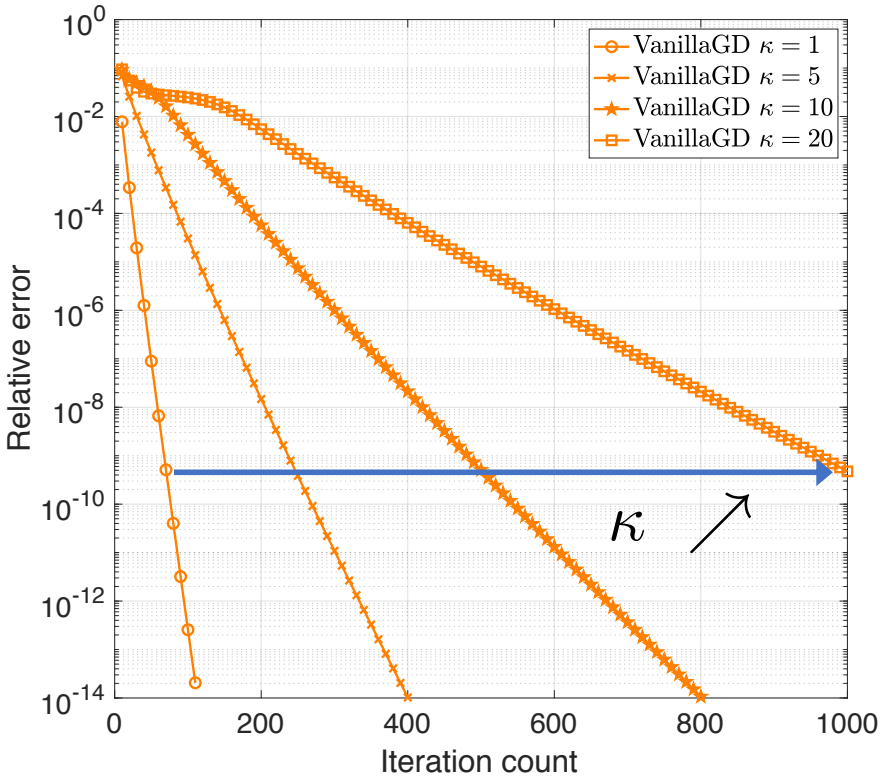
$$m \gtrsim (n_1 + n_2) r^2 \kappa^2.$$

Similar results hold for many low-rank problems.

(Netrapalli et al. '13, Candès, Li, Soltanolkotabi '14, Sun and Luo '15, Chen and Wainwright '15, Zheng and Lafferty '15, Ma et al. '17, ...)

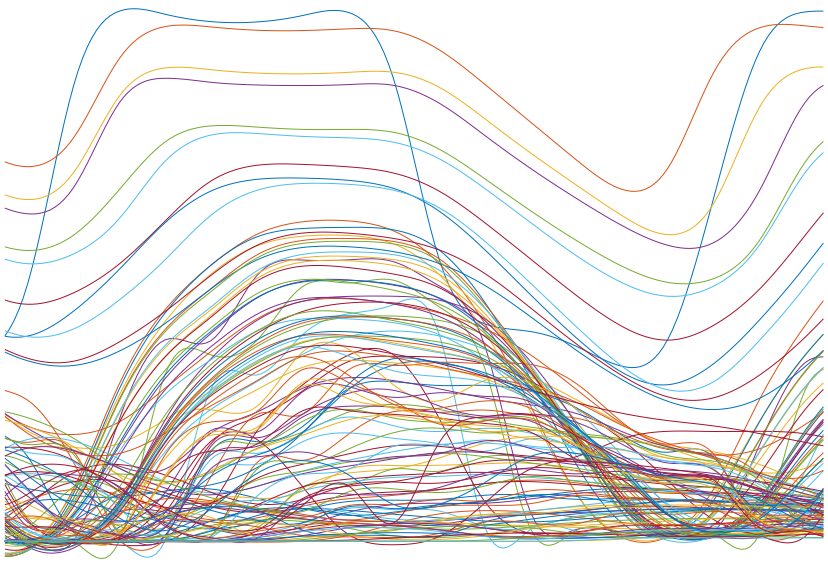
Convergence slows down for ill-conditioned matrices

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad f(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\left\|\mathcal{P}_\Omega(\boldsymbol{X}\boldsymbol{Y}^\top - \boldsymbol{M})\right\|_{\mathrm{F}}^2$$

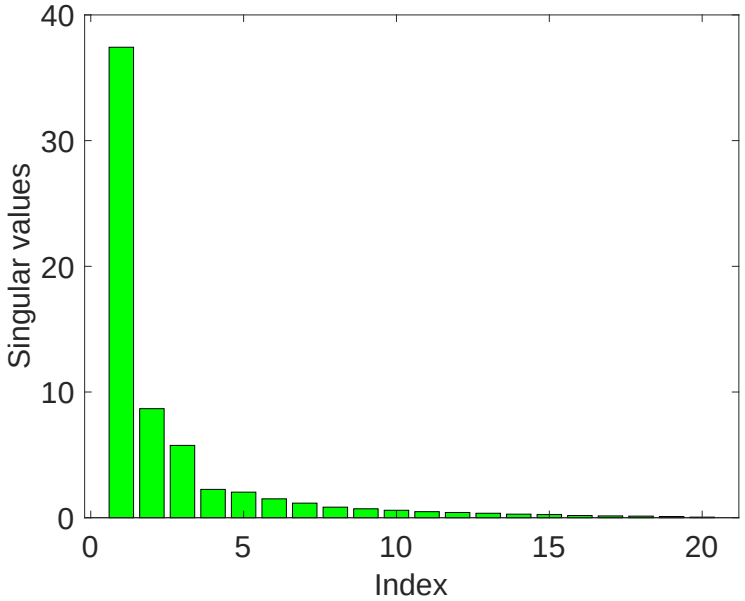


Vanilla GD converges in $O\left(\kappa \log \frac{1}{\varepsilon}\right)$ iterations.

Condition number can be large



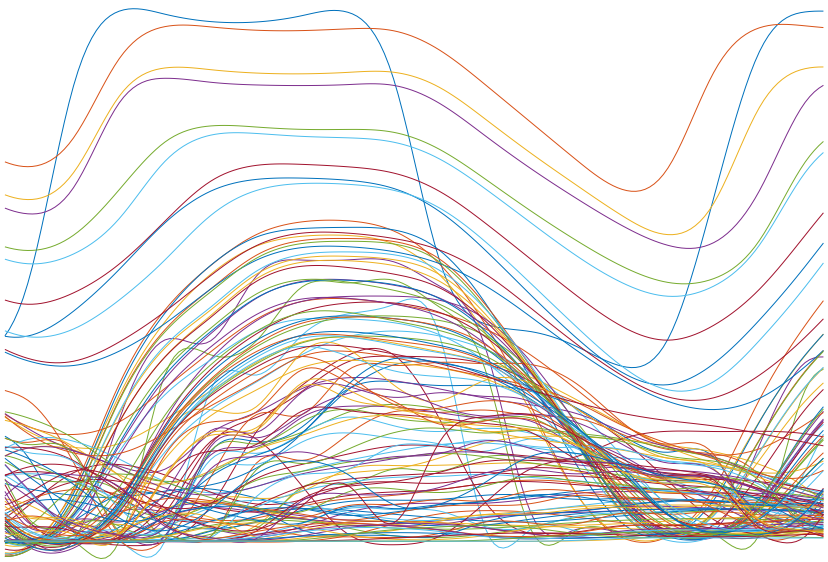
chlorine concentration levels
120 junctions, 180 time slots



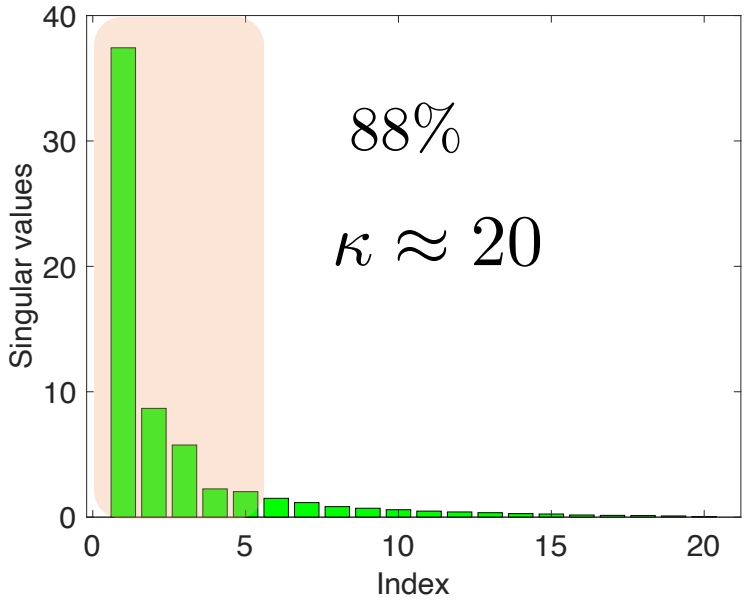
power-law spectrum

Data source: `www.epa.gov/water-research/epanet`

Condition number can be large



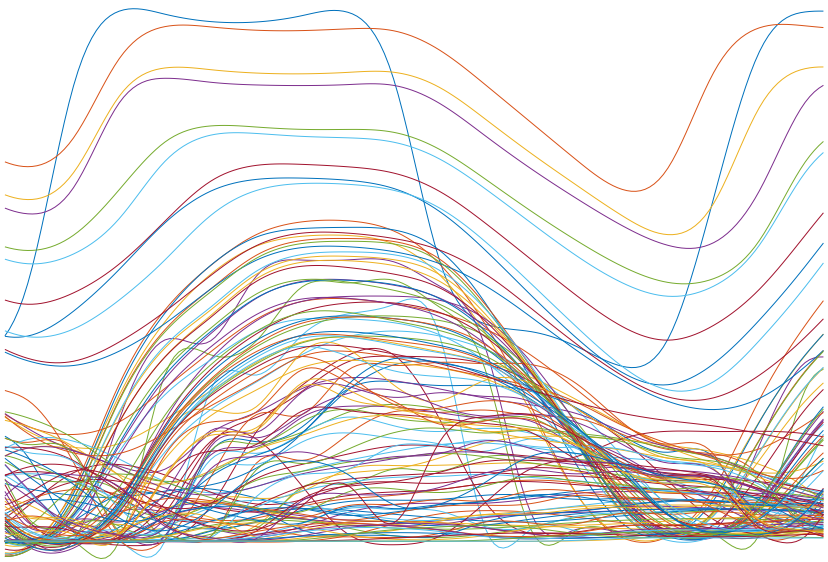
chlorine concentration levels
120 junctions, 180 time slots



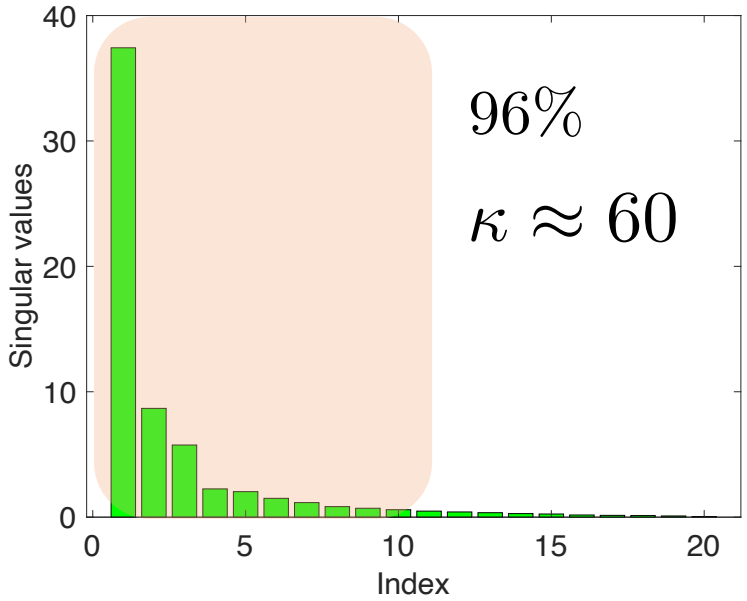
rank-5 approximation

Data source: `www.epa.gov/water-research/epanet`

Condition number can be large



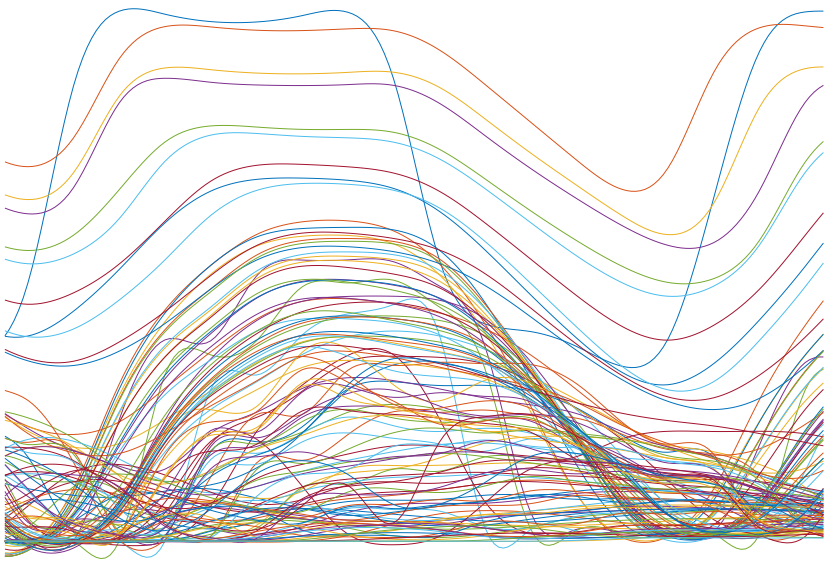
chlorine concentration levels
120 junctions, 180 time slots



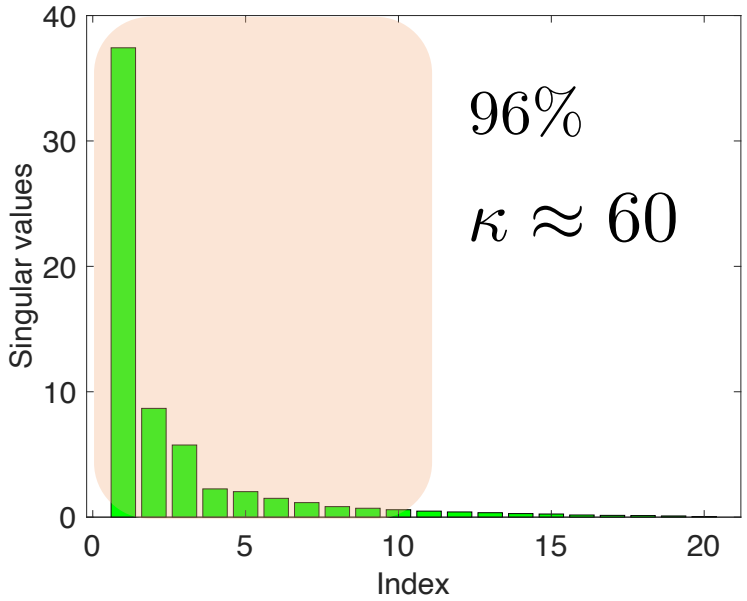
rank-10 approximation

Data source: `www.epa.gov/water-research/epanet`

Condition number can be large



chlorine concentration levels
120 junctions, 180 time slots



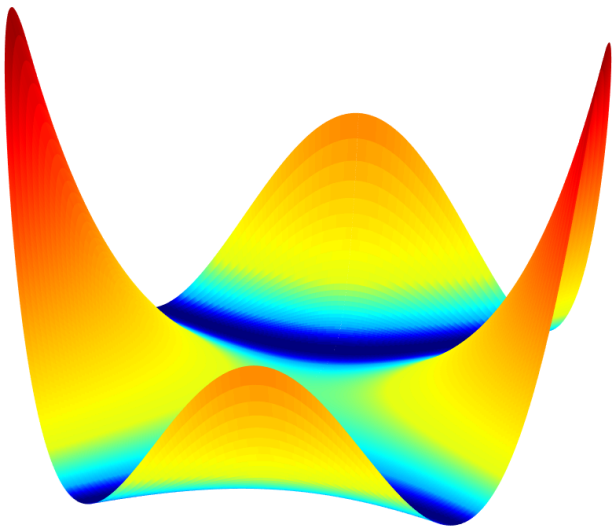
rank-10 approximation

Can we accelerate the convergence rate of GD to $O(\log \frac{1}{\varepsilon})$?

Data source: `www.epa.gov/water-research/epanet`

A new algorithm: scaled gradient descent (ScaledGD)

$$f(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_2^2$$



- **Spectral initialization:** find an initial point in the "basin of attraction".
- **Scaled gradient iterations:**

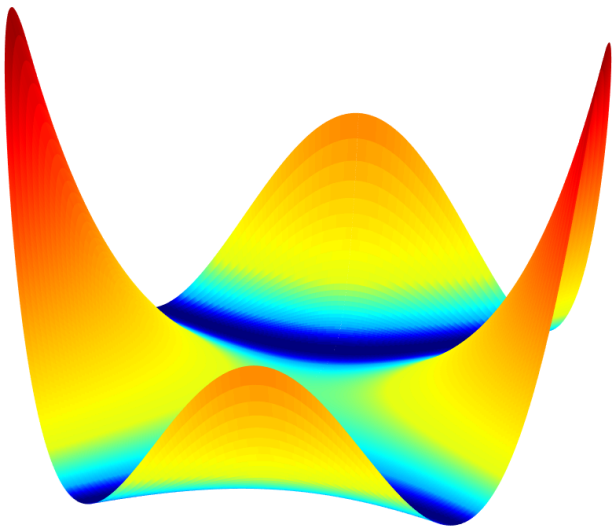
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta \nabla_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1}}_{\texttt{preconditioner}}$$

for $t = 0, 1, \ldots$

A new algorithm: scaled gradient descent (ScaledGD)

$$f(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_2^2$$



- **Spectral initialization:** find an initial point in the "basin of attraction".
- **Scaled gradient iterations:**

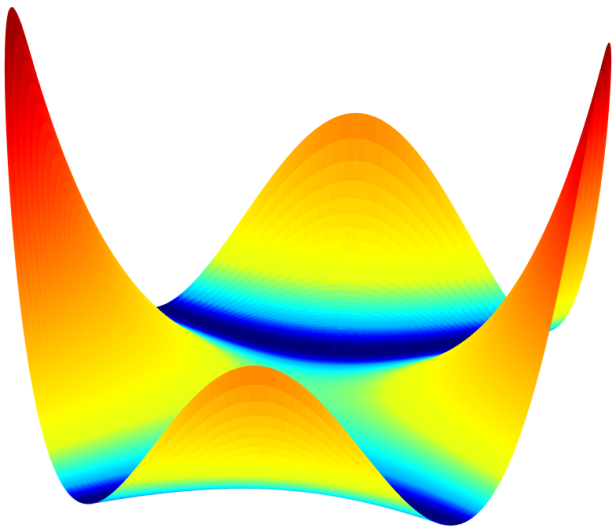
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta \nabla_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1}}_{\texttt{preconditioner}}$$

for $t = 0, 1, \ldots$

A new algorithm: scaled gradient descent (ScaledGD)

$$f(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\left\|\boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top)\right\|_2^2$$



- **Spectral initialization:** find an initial point in the "basin of attraction".
- **Scaled gradient iterations:**

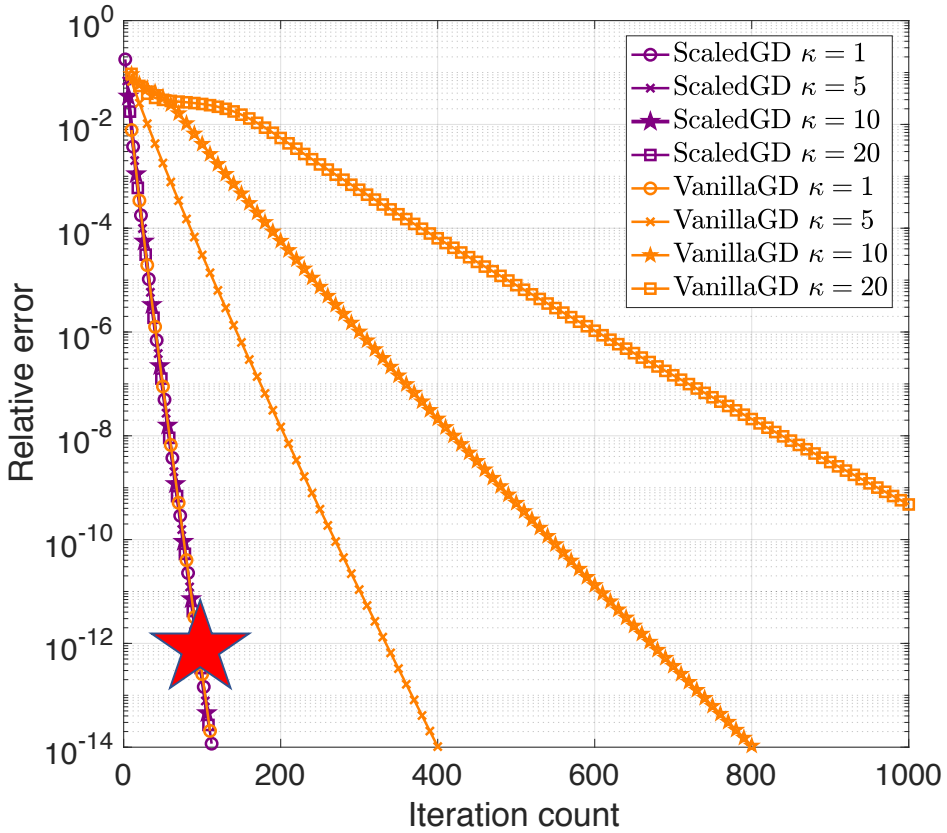
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta \nabla_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1}}_{\texttt{preconditioner}}$$

for $t = 0, 1, \ldots$

ScaledGD is a *preconditioned* gradient method *without* balancing regularization!

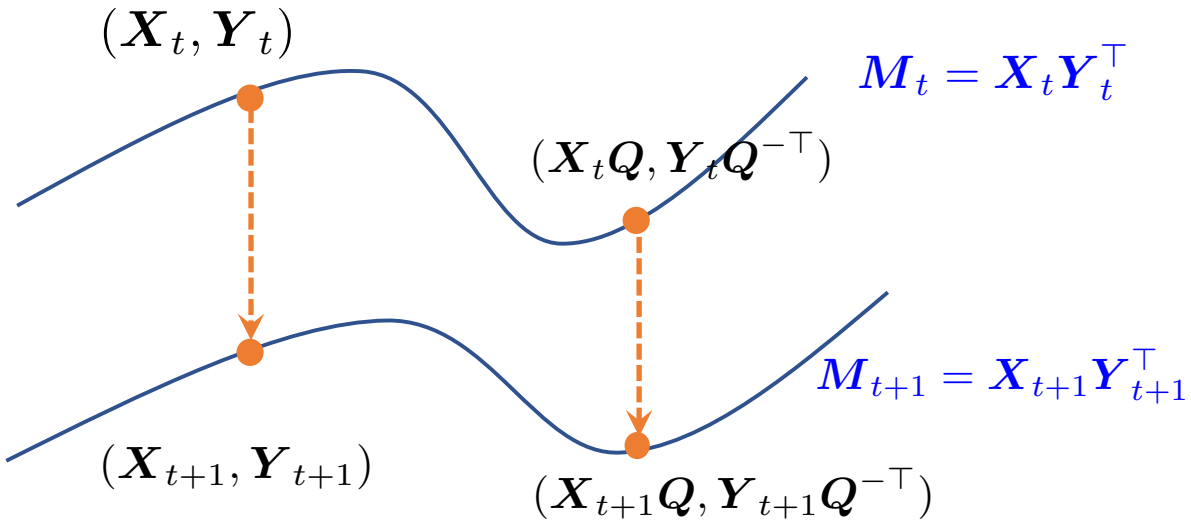
ScaledGD for low-rank matrix completion



Huge computational saving: ScaledGD converges in an κ-independent manner with a minimal overhead!

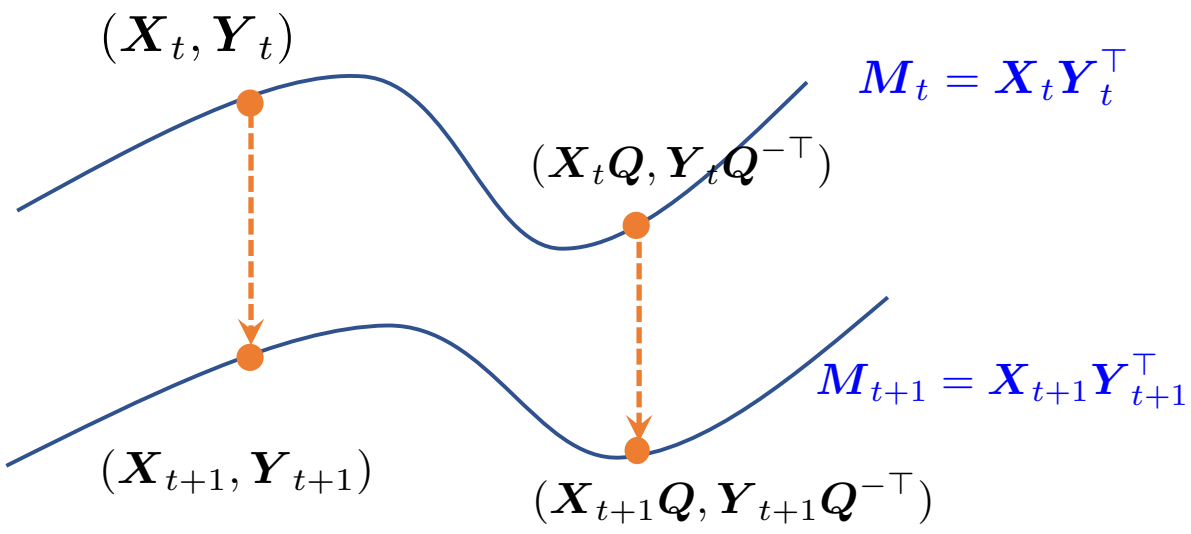
A closer look at ScaledGD

Invariance to invertible transforms: (Tanner and Wei, '16; Mishra '16)



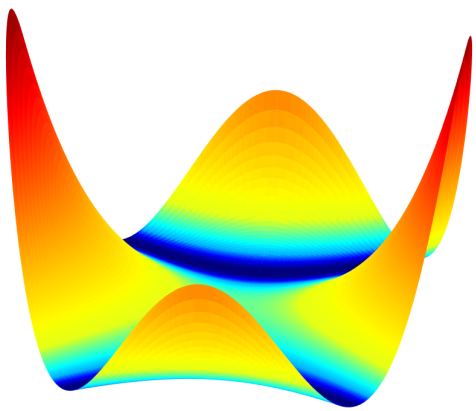
A closer look at ScaledGD

Invariance to invertible transforms: (Tanner and Wei, '16; Mishra '16)



New distance metric as Lyapunov function:

$$\operatorname{dist}^2\left(\begin{bmatrix}\boldsymbol{X}\\ \boldsymbol{Y}\end{bmatrix}, \begin{bmatrix}\boldsymbol{X}_\star\\ \boldsymbol{Y}_\star\end{bmatrix}\right) = \inf_{\boldsymbol{Q}\in \mathrm{GL}(r)} \left\|(\boldsymbol{X}\boldsymbol{Q} - \boldsymbol{X}_\star)\boldsymbol{\Sigma}_\star^{1/2}\right\|_{\mathrm{F}}^2 + \left\|(\boldsymbol{Y}\boldsymbol{Q}^{-\top} - \boldsymbol{Y}_\star)\boldsymbol{\Sigma}_\star^{1/2}\right\|_{\mathrm{F}}^2$$



$+$ a careful trajectory-based analysis

Theoretical guarantees of ScaledGD

Theorem (Tong, Ma and Chi, 2020)

For low-rank matrix sensing with i.i.d. Gaussian design, ScaledGD with spectral initialization achieves

$$\|\boldsymbol{X}_t\boldsymbol{Y}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \lesssim \varepsilon \cdot \sigma_{\min}(\boldsymbol{M})$$

- **Computational:** *within* $O\big(\log \frac{1}{\varepsilon}\big)$ *iterations;*
- **Statistical:** *the sample complexity satisfies*

$$m \gtrsim (n_1 + n_2) r^2 \kappa^2.$$

Theoretical guarantees of ScaledGD

Theorem (Tong, Ma and Chi, 2020)

For low-rank matrix sensing with i.i.d. Gaussian design, ScaledGD with spectral initialization achieves

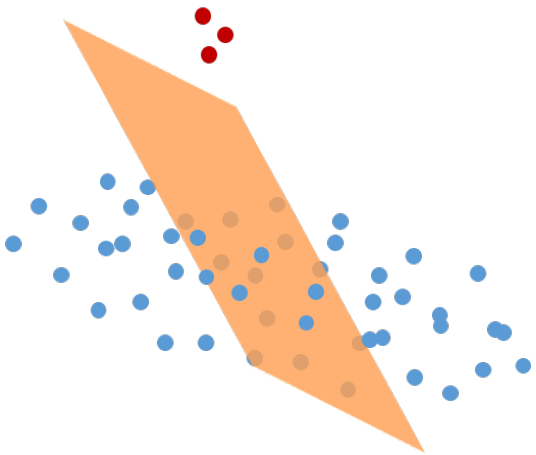
$$\|\boldsymbol{X}_t\boldsymbol{Y}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \lesssim \varepsilon \cdot \sigma_{\min}(\boldsymbol{M})$$

- **Computational:** *within* $O\big(\log \frac{1}{\varepsilon}\big)$ *iterations;*
- **Statistical:** *the sample complexity satisfies*

$$m \gtrsim (n_1 + n_2) r^2 \kappa^2.$$

Strict improvement over Tu et al.: ScaledGD provably accelerates vanilla GD at the same sample complexity!

ScaledGD works more broadly



$$\begin{bmatrix} \checkmark & ? & ? & ? & \checkmark \\ ? & ? & \checkmark & \checkmark & ? \\ \checkmark & ? & ? & \checkmark & ? \\ ? & ? & \checkmark & ? & ? \\ \checkmark & ? & ? & ? & ? \\ ? & \checkmark & ? & ? & \checkmark \end{bmatrix}$$

	Robust PCA		**Matrix completion**	
Algorithms	corruption fraction	iteration complexity	sample complexity	iteration complexity
GD	$\frac{1}{\mu r^{3/2}\kappa^{3/2} \vee \mu r \kappa^2}$	$\kappa \log \frac{1}{\varepsilon}$	$(\mu \vee \log n)\mu n r^2 \kappa^2$	$\kappa \log \frac{1}{\varepsilon}$
ScaledGD	$\frac{1}{\mu r^{3/2}\kappa}$	$\log \frac{1}{\varepsilon}$	$(\mu \kappa^2 \vee \log n)\mu n r^2 \kappa^2$	$\log \frac{1}{\varepsilon}$

Huge computation savings at comparable sample complexities!

Code available at `https://github.com/Titan-Tong/ScaledGD`

What about the run time?

The run time of ScaledGD is rather competitive, with additional suitability for parallel implementation.

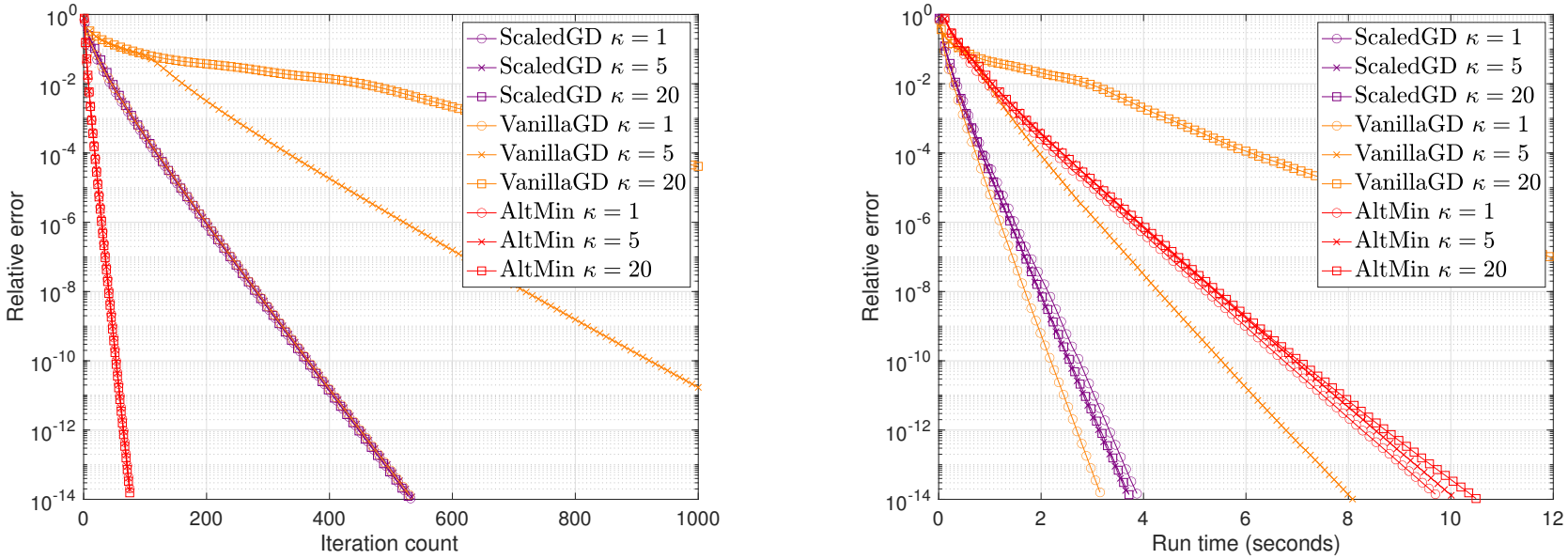
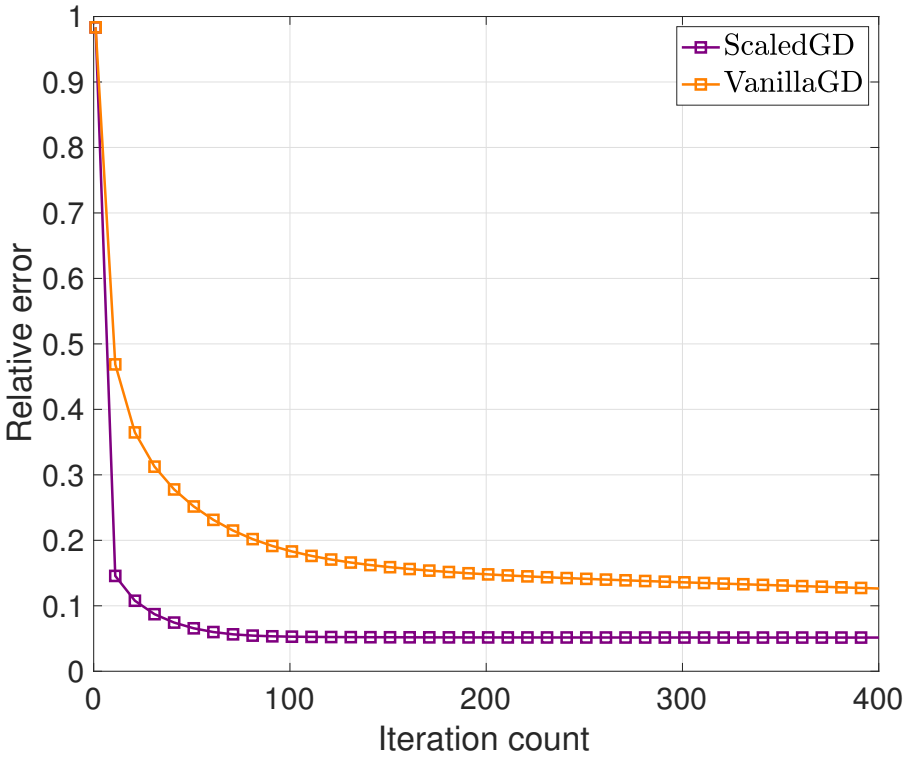


Figure: Run time for matrix completion with $n = 1000$, $p = 0.2$, $r = 50$.

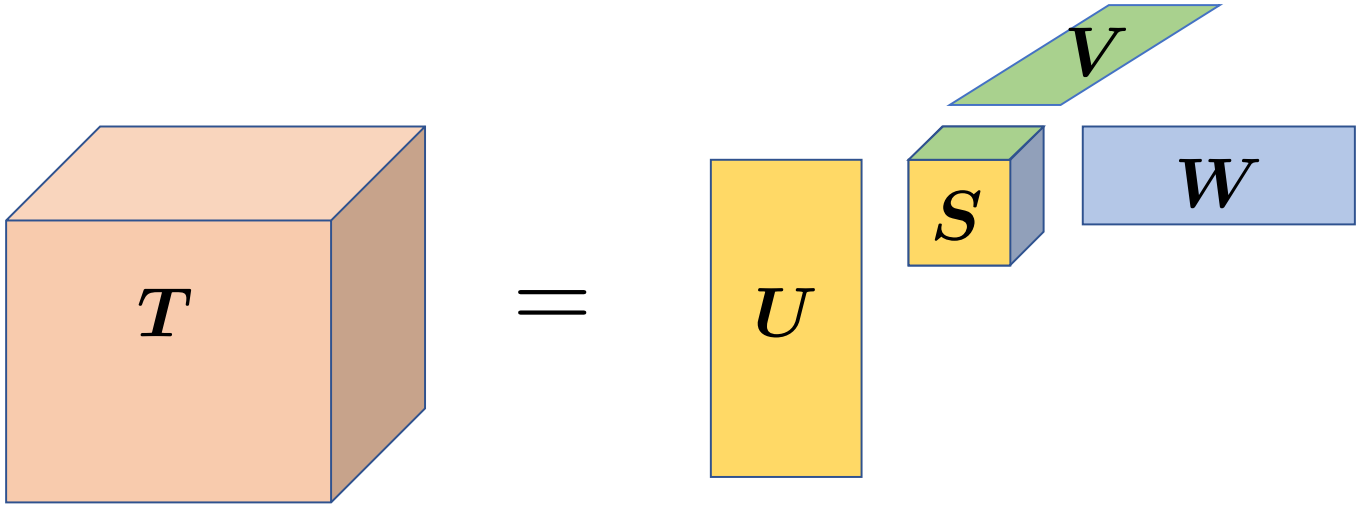
Numerical stability

ScaledGD converges faster than vanilla GD in a small number of iterations (they eventually reach the same accuracy).



Generalization to tensors

Low-rank tensor under Tucker decomposition



Low-rank Tucker decomposition of a tensor:

$$\boldsymbol{T} = (\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W}) \cdot \boldsymbol{S},$$

where $\boldsymbol{U} \in \mathbb{R}^{n_1 \times r_1}$, $\boldsymbol{V} \in \mathbb{R}^{n_2 \times r_2}$, $\boldsymbol{W} \in \mathbb{R}^{n_3 \times r_3}$ and $\boldsymbol{S} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$.

Applications in fMRI imaging, recommendation systems, etc...

ScaledGD for ill-conditioned low-rank tensor estimation

$$\min_{\boldsymbol{F}=(\boldsymbol{U},\boldsymbol{V},\boldsymbol{W},\boldsymbol{S})} f(\boldsymbol{F}) = \frac{1}{2}\|\mathcal{A}((\boldsymbol{U},\boldsymbol{V},\boldsymbol{W})\cdot\boldsymbol{S}) - \boldsymbol{y}\|_2^2$$

Scaled gradient iterations:

$$\begin{aligned}
\boldsymbol{U}_{t+1} &= \boldsymbol{U}_t - \eta\nabla_{\boldsymbol{U}} f(\boldsymbol{F}_t)\big(\breve{\boldsymbol{U}}_t^\top\breve{\boldsymbol{U}}_t\big)^{-1},\\
\boldsymbol{V}_{t+1} &= \boldsymbol{V}_t - \eta\nabla_{\boldsymbol{V}} f(\boldsymbol{F}_t)\big(\breve{\boldsymbol{V}}_t^\top\breve{\boldsymbol{V}}_t\big)^{-1},\\
\boldsymbol{W}_{t+1} &= \boldsymbol{W}_t - \eta\nabla_{\boldsymbol{W}} f(\boldsymbol{F}_t)\big(\breve{\boldsymbol{W}}_t^\top\breve{\boldsymbol{W}}_t\big)^{-1},\\
\boldsymbol{S}_{t+1} &= \boldsymbol{S}_t - \eta\left((\boldsymbol{U}_t^\top\boldsymbol{U}_t)^{-1},(\boldsymbol{V}_t^\top\boldsymbol{V}_t)^{-1},(\boldsymbol{W}_t^\top\boldsymbol{W}_t)^{-1}\right)\cdot\nabla_{\boldsymbol{S}} f(\boldsymbol{F}_t),
\end{aligned}$$

where $\breve{\boldsymbol{U}}_t := (\boldsymbol{V}_t\otimes\boldsymbol{W}_t)\mathcal{M}_1(\boldsymbol{S}_t)^\top$, $\breve{\boldsymbol{V}}_t := (\boldsymbol{U}_t\otimes\boldsymbol{W}_t)\mathcal{M}_2(\boldsymbol{S}_t)^\top$, and $\breve{\boldsymbol{W}}_t := (\boldsymbol{U}_t\otimes\boldsymbol{V}_t)\mathcal{M}_3(\boldsymbol{S}_t)^\top$. Here, $\mathcal{M}_k(\boldsymbol{S})$ is the matricization of $\boldsymbol{S}$ along the k-th mode.

Key property: invariance to parameterization.

ScaledGD for low-rank tensor completion

Theorem (Tong et. al., 2021)

For low-rank tensor completion under Bernoulli sampling, assume $n = n_1 = n_2 = n_3$, *ScaledGD with spectral initialization and projection achieves*

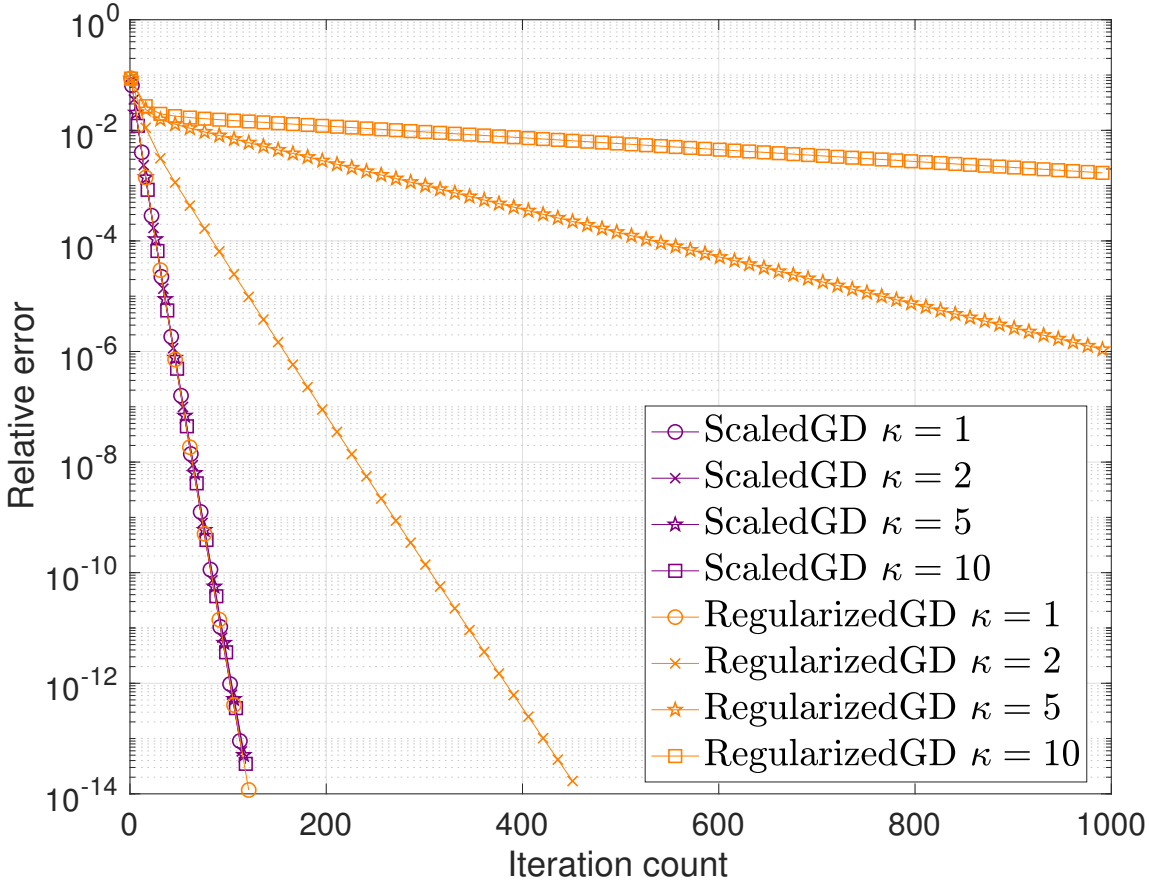
$$\|(\boldsymbol{U}_t,\, \boldsymbol{V}_t,\, \boldsymbol{W}_t) \cdot \boldsymbol{S}_t - \boldsymbol{T}\|_{\mathrm{F}} \lesssim \varepsilon \cdot \sigma_{\min}(\boldsymbol{T})$$

- **Computational:** *within* $O\big(\log \frac{1}{\varepsilon}\big)$ *iterations;*
- **Statistical:** *as long as the sample complexity satisfies*

$$n^3 p \gtrsim \mu^{3/2} r^{5/2} n^{3/2} \kappa^3 \log n.$$

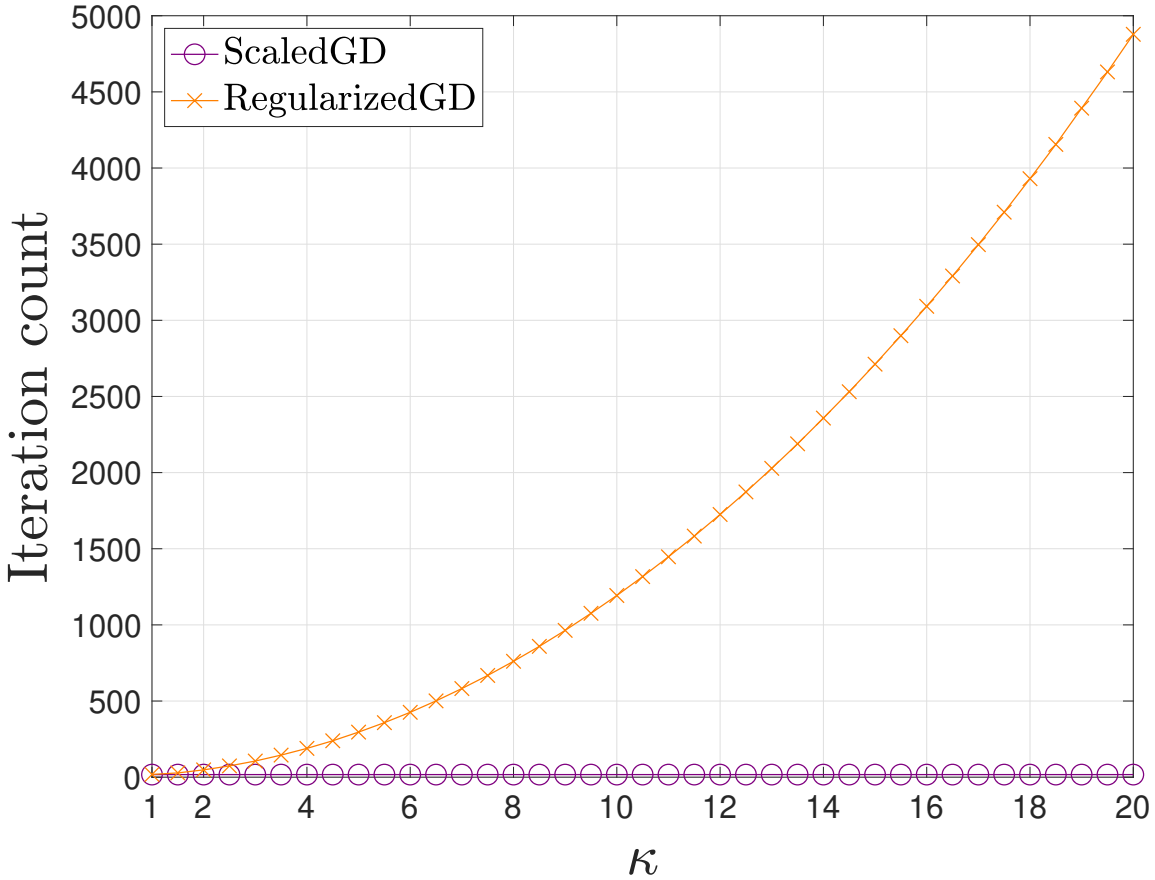
First provable linear convergence at a near-optimal sample complexity for low-Tucker-rank tensor completion!

Numerical evidence



The benefit of ScaledGD is even more evident for tensors!

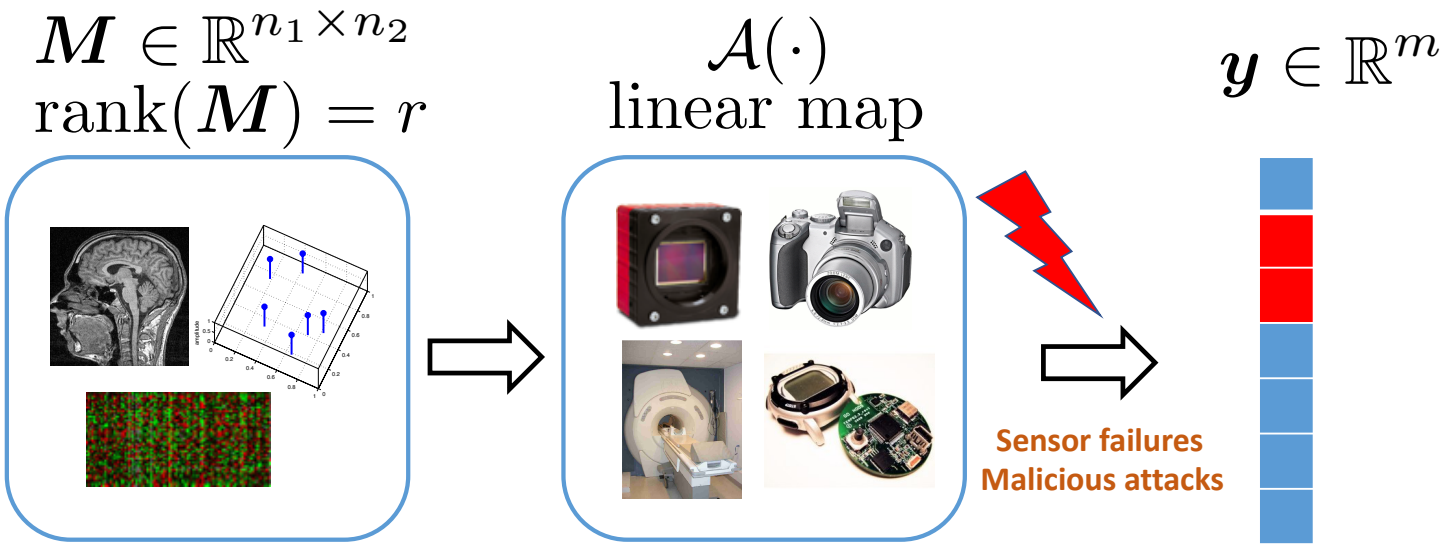
Numerical evidence



The benefit of ScaledGD is even more evident for tensors!

Robustness to outliers and corruptions?

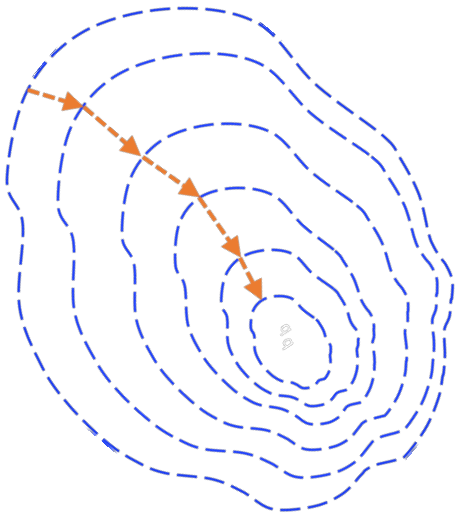
Outlier-corrupted low-rank matrix sensing



$$\boldsymbol{y} \;=\; \mathcal{A}(\boldsymbol{M}) + \underbrace{\boldsymbol{s}}_{\text{outliers}}, \quad \mathcal{A}(\boldsymbol{M}) = \{\langle \boldsymbol{A}_i, \boldsymbol{M} \rangle\}_{i=1}^m$$

Arbitrary but sparse outliers: $\|\boldsymbol{s}\|_0 \leq \alpha \cdot m$, where $0 \leq \alpha < 1$ is fraction of outliers.

Existing approaches fail



- **Spectral initialization would fail:**
  $\boldsymbol{X}_0 \leftarrow$ top-r SVD of

$$\boldsymbol{Y} = \frac{1}{m} \sum_{i=1}^{m} y_i \boldsymbol{A}_i$$

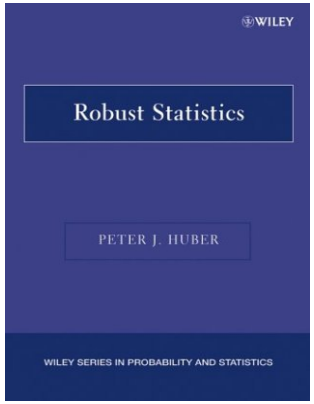
- **Gradient iterations would fail:**

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \frac{\eta}{m} \sum_{i=1}^{m} \nabla \ell_i(y_i; \boldsymbol{X}_t)$$

for $t = 0, 1, \ldots$

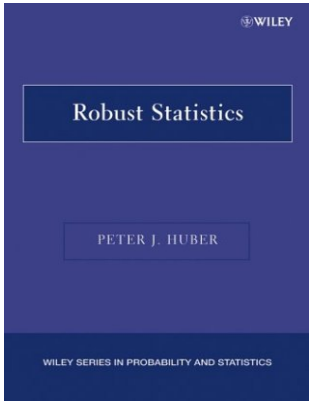
Even a single outlier can fail the algorithm!

Median-truncated gradient descent



Key idea: "median-truncation" — discard samples *adaptively* based on how large sample gradients / values deviate from median

Median-truncated gradient descent

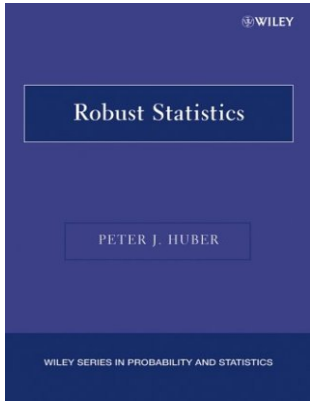


Key idea: "median-truncation" — discard samples *adaptively* based on how large sample gradients / values deviate from median

- **Robustify spectral initialization:** $\boldsymbol{X}_0 \leftarrow$ top-r SVD of

$$\boldsymbol{Y} = \frac{1}{m} \sum_{i:|y_i| \lesssim \text{median}(|y_i|)} y_i \boldsymbol{A}_i$$

Median-truncated gradient descent



Key idea: "median-truncation" — discard samples *adaptively* based on how large sample gradients / values deviate from median

- **Robustify spectral initialization:** $\boldsymbol{X}_0 \leftarrow$ top-r SVD of

$$\boldsymbol{Y} = \frac{1}{m} \sum_{i:|y_i| \lesssim \text{median}(|y_i|)} y_i \boldsymbol{A}_i$$

- **Robustify gradient descent:**

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \frac{\eta}{m} \sum_{i:|r_t^i| \lesssim \text{median}(|r_t^i|)} \nabla \ell_i(y_i; \boldsymbol{X}_t), \quad t = 0, 1, \ldots$$

where $r_t^i := \left| y_i - \langle \boldsymbol{A}_i, \boldsymbol{X}_t \boldsymbol{X}_t^\top \rangle \right|$ is the size of the gradient.

Theoretical guarantees

Theorem (Li, Chi, Zhang, and Liang, IMIAI 2020)

For low-rank matrix sensing with i.i.d. Gaussian design, median-truncated GD (with robust spectral initialization) achieves

$$\|\boldsymbol{X}_t\boldsymbol{X}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \le \varepsilon \cdot \sigma_{\min}(\boldsymbol{M}),$$

- **Computational:** *within* $O\left(\kappa \log \frac{1}{\varepsilon}\right)$ *iterations;*
- **Statistical:** *the sample complexity satisfies*

$$m \gtrsim nr^2 \mathsf{poly}(\kappa, \log n);$$

- **Robustness:** *and the fraction of outliers*

$$\alpha \lesssim 1/\sqrt{r}.$$

Theoretical guarantees

Theorem (Li, Chi, Zhang, and Liang, IMIAI 2020)

For low-rank matrix sensing with i.i.d. Gaussian design, median-truncated GD (with robust spectral initialization) achieves

$$\|\boldsymbol{X}_t\boldsymbol{X}_t^\top - \boldsymbol{M}\|_{\mathrm{F}} \le \varepsilon \cdot \sigma_{\min}(\boldsymbol{M}),$$

- **Computational:** *within* $O\left(\kappa \log \frac{1}{\varepsilon}\right)$ *iterations;*
- **Statistical:** *the sample complexity satisfies*

$$m \gtrsim nr^2 \mathsf{poly}(\kappa, \log n);$$

- **Robustness:** *and the fraction of outliers*

$$\alpha \lesssim 1/\sqrt{r}.$$

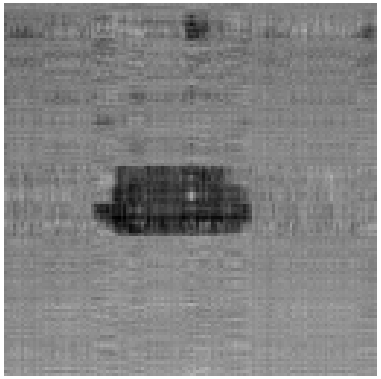
Median-truncated GD adds robustness to GD *obliviously*.

Numerical example

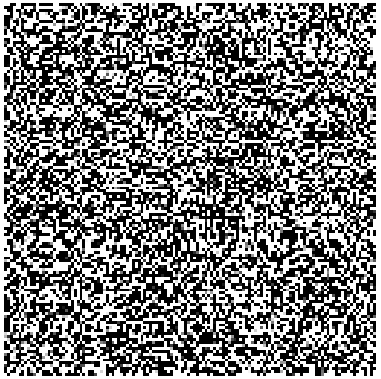
Low-rank matrix sensing:

$$y_i = \langle \boldsymbol{A}_i, \boldsymbol{M} \rangle + s_i, \quad i = 1, \ldots, m$$

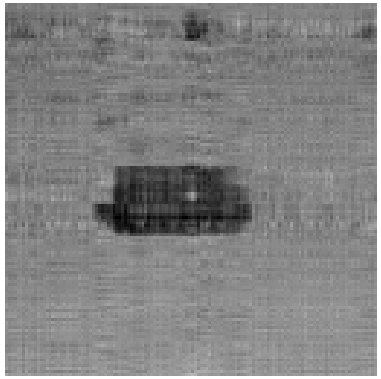
Ground truth



GD
no outliers



GD
1% outliers



median-TGD
1% outliers

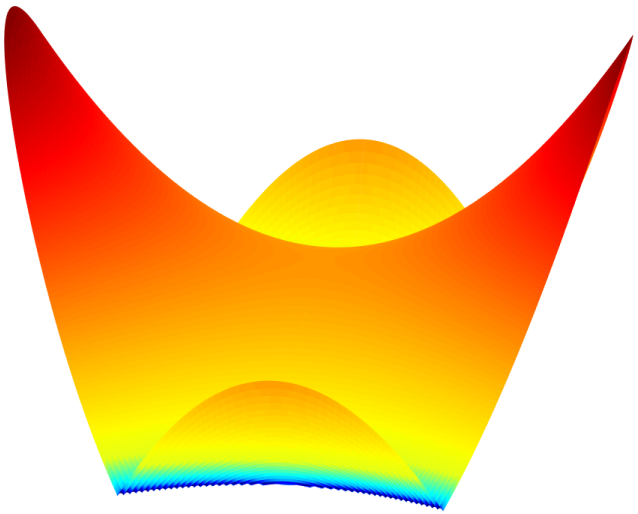
Median-truncated GD achieves similar performance as if performing GD on the clean data.

Li, Chi, Zhang and Liang, "Non-convex low-rank matrix recovery with arbitrary outliers via median-truncated gradient descent", Information and Inference: A Journal of the IMA, 2020.

Dealing with outliers: subgradient methods

Least absolute deviation (LAD): (Charisopoulos et.al.'19; Li et al'18)

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad f(\boldsymbol{X},\boldsymbol{Y}) = \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_1$$



Subgradient iterations:

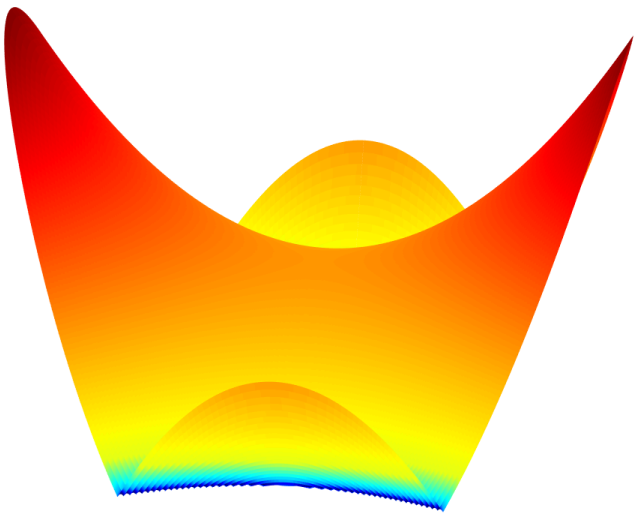
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta_t \, \partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)$$
$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta_t \, \partial_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)$$

where η_t is set as Polyak's or geometric decaying stepsize.

Dealing with outliers: scaled subgradient methods

Least absolute deviation (LAD):

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad f(\boldsymbol{X},\boldsymbol{Y}) = \left\| \boldsymbol{y} - \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top) \right\|_1$$



Scaled subgradient iterations:

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta_t \, \partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta_t \, \partial_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1}}_{\texttt{preconditioner}}$$

where η_t is set as Polyak's or geometric decaying stepsize.

Stepsize schedule

Polyak's stepsize:

$$\eta_t = \frac{f(\boldsymbol{X}_t \boldsymbol{Y}_t^\top) - f(\boldsymbol{M})}{\|\partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1/2}\|_{\mathrm{F}}^2 + \|\partial_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1/2}\|_{\mathrm{F}}^2}.$$

- Use the distance concerted with preconditioners.
- Require the knowledge of the optimal value $f(\boldsymbol{X}_\star)$.

Stepsize schedule

Polyak's stepsize:

$$\eta_t = \frac{f(\boldsymbol{X}_t \boldsymbol{Y}_t^\top) - f(\boldsymbol{M})}{\|\partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1/2}\|_{\mathrm{F}}^2 + \|\partial_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1/2}\|_{\mathrm{F}}^2}.$$

- Use the distance concerted with preconditioners.
- Require the knowledge of the optimal value $f(\boldsymbol{X}_\star)$.

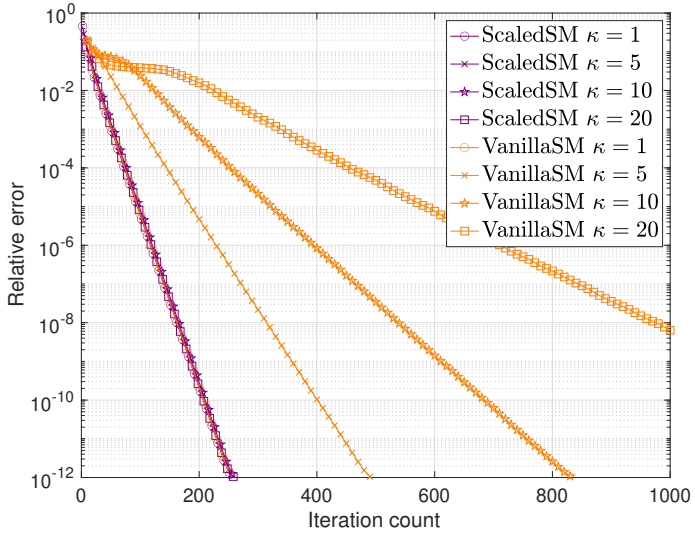
Geometrically decaying stepsize:

$$\eta_t = \frac{\lambda q^t}{\sqrt{\|\partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1/2}\|_{\mathrm{F}}^2 + \|\partial_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)(\boldsymbol{X}_t^\top \boldsymbol{X}_t)^{-1/2}\|_{\mathrm{F}}^2}}$$

- Parameters λ, q need to be tuned.
- Perform similarly as Polyak's stepsize under well-tuned λ, q.

Performance guarantees

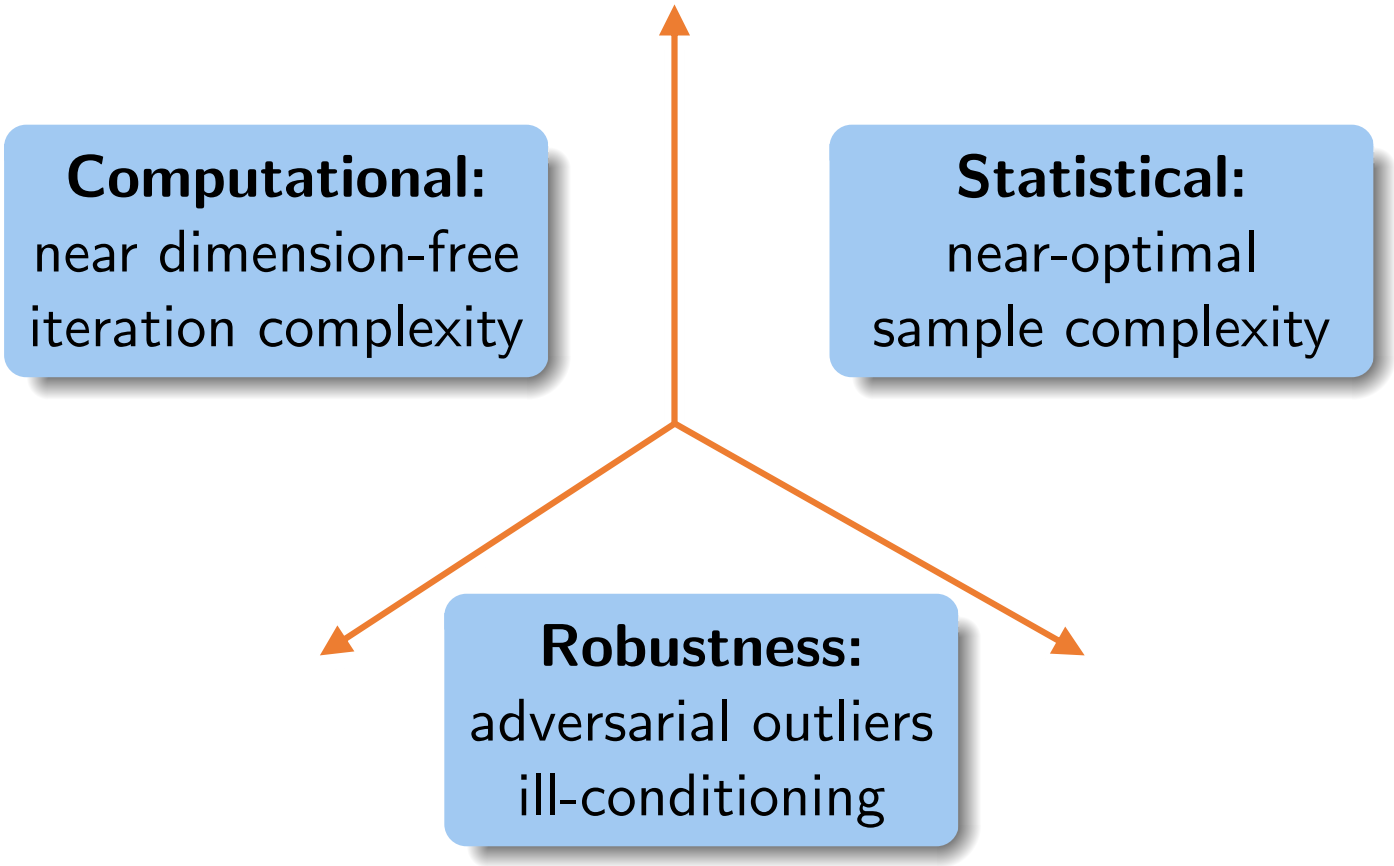
	matrix sensing	**quadratic sensing**
Subgradient Method (Charisopoulos et al, '19)	$\frac{\kappa}{(1-2\alpha)^2} \log \frac{1}{\epsilon}$	$\frac{r\kappa}{(1-2\alpha)^2} \log \frac{1}{\epsilon}$
ScaledSM (Tong, Ma, Chi, '20)	$\frac{1}{(1-2\alpha)^2} \log \frac{1}{\epsilon}$	$\frac{r}{(1-2\alpha)^2} \log \frac{1}{\epsilon}$



Robustness to both ill-conditioning and adversarial corruptions!

Concluding remarks

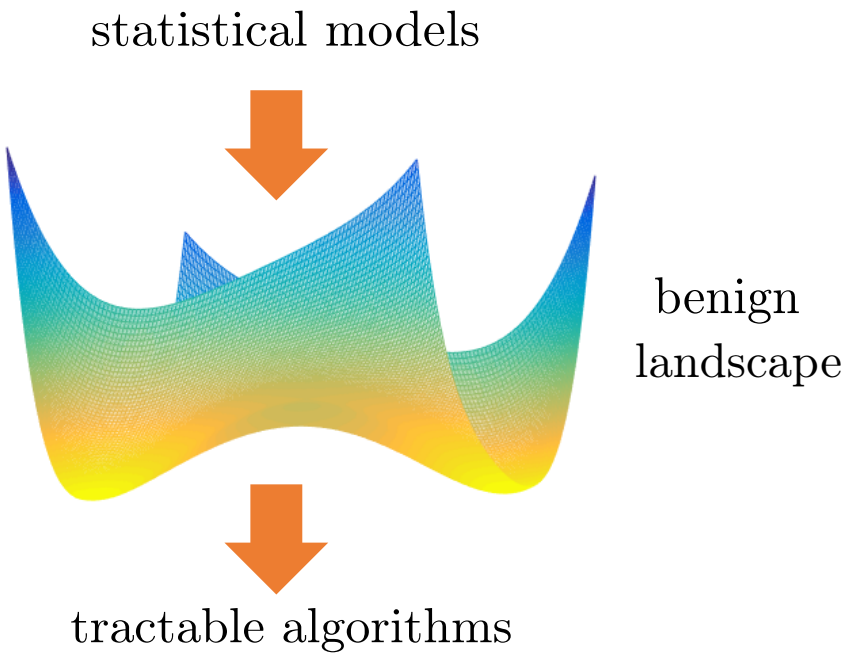
Bridging the theory-practice gap



Nonconvex low-rank matrix estimation:

- identification and exploitation of benign geometric properties;
- analyzing iterate trajectories beyond black-box optimization;
- simple variants of GD lead to robust and accelerated convergence.

Statistical thinking + Optimization efficiency



When data are generated by certain statistical models, problems are often much nicer than worst-case instances

A growing list of "benign" nonconvex problems

- phase retrieval
- matrix sensing
- matrix completion
- blind deconvolution / self-calibration
- dictionary learning
- tensor decomposition / completion
- robust PCA
- mixed linear regression
- learning one-layer neural networks
- ...

Selected References

Overview:

1. Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview, Y. Chi, Y. M. Lu and Y. Chen, *IEEE Trans. on Signal Processing*, 2019.
2. Spectral Methods for Data Science: A Statistical Perspective", Y. Chen, Y. Chi, J. Fan and C. Ma, *Foundations and Trends in Machine Learning*, 2021.
3. Harnessing Structures in Big Data via Guaranteed Low-Rank Matrix Estimation, Y. Chen and Y. Chi, *IEEE Signal Processing Magazine*, 2018.

Geometry of factored gradient descent:

1. Implicit Regularization for Nonconvex Statistical Estimation, C. Ma, K. Wang, Y. Chi and Y. Chen, *Foundations of Computational Mathematics*, 2020.
2. Beyond Procrustes: Balancing-free Gradient Descent for Asymmetric Low-Rank Matrix Sensing, C. Ma, Y. Li and Y. Chi, *IEEE Trans. on Signal Processing*, 2021.
3. Nonconvex Matrix Factorization from Rank-One Measurements, Y. Li, C. Ma, Y. Chen, and Y. Chi, *IEEE Trans. on Information Theory*, 2020.

Selected References

Robustness to ill-conditioning:

1. Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent, T. Tong, C. Ma, and Y. Chi, *Journal of Machine Learning Research*, 2021.
2. Scaling and scalability: Provable nonconvex low-rank tensor estimation from incomplete measurements, T. Tong, C. Ma, A. Prater-Bennette, E. Tripp, and Y. Chi, *arXiv preprint arXiv:2104.14526*, 2021.

Robustness to adversarial outliers:

1. Low-rank matrix recovery with scaled subgradient methods: Fast and robust convergence without the condition number, T. Tong, C. Ma, and Y. Chi, *IEEE Trans. on Signal Processing*, 2021.
2. Non-convex low-rank matrix recovery with arbitrary outliers via median-truncated gradient descent, Y. Li, Y. Chi, H. Zhang and Y. Liang, *Information and Inference: A Journal of the IMA*, 2020.

Thanks!

https://users.ece.cmu.edu/~yuejiec/