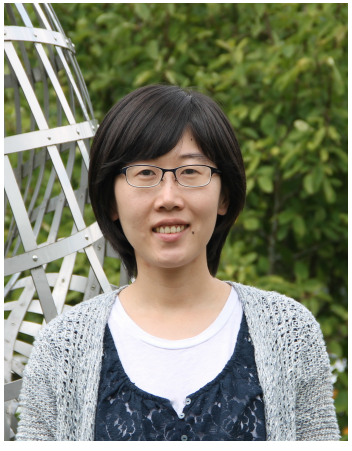
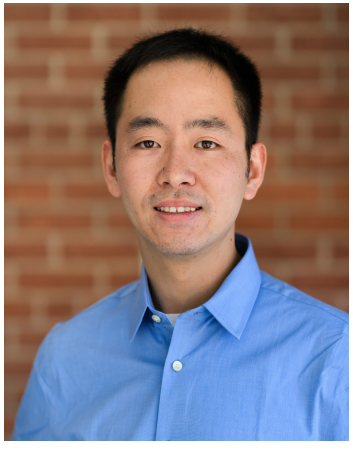
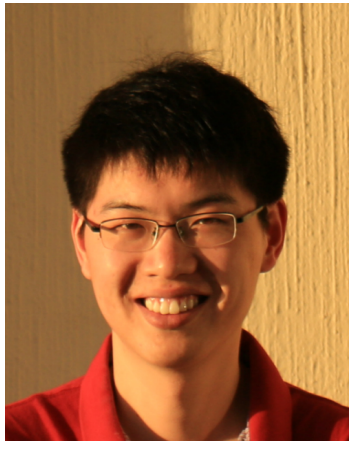
Nonconvex Optimization for High-Dimensional Signal Estimation: Spectral and Iterative Methods – Part I



Yuejie Chi
Carnegie Mellon



Yuxin Chen
Princeton



Cong Ma
UC Berkeley

EUSIPCO Tutorial, December 2020

Acknowledgement

- Our students and collaborators: Emmanuel J. Candès, Jianqing Fan, Yuanxin Li, Yingbin Liang, Yue M. Lu, Laixi Shi, Vincent Monardo, Tian Tong, Kaizheng Wang, Huishuai Zhang.

- This work is supported in part by ARO, AFOSR, ONR and NSF.

Outline

- Part I: Introduction and Warm-Up

  Why nonconvex? basic concepts and a warm-up example (PCA)

- Part II: Gradient Descent and Implicit Regularization

  phase retrieval, matrix completion, random initialization

- Part III: Spectral Methods

  a general recipe, ℓ_2 and ℓ_∞ guarantees, community detection

- Part IV: Robustness to Corruptions and Ill-Conditioning

  median truncation, least absolute deviation, scaled gradient descent

Outline

- Part I: Introduction and Warm-Up

  Why nonconvex? basic concepts and a warm-up example (PCA)

- Part II: Gradient Descent and Implicit Regularization

  phase retrieval, matrix completion, random initialization

- Part III: Spectral Methods

  a general recipe, ℓ_2 and ℓ_∞ guarantees, community detection

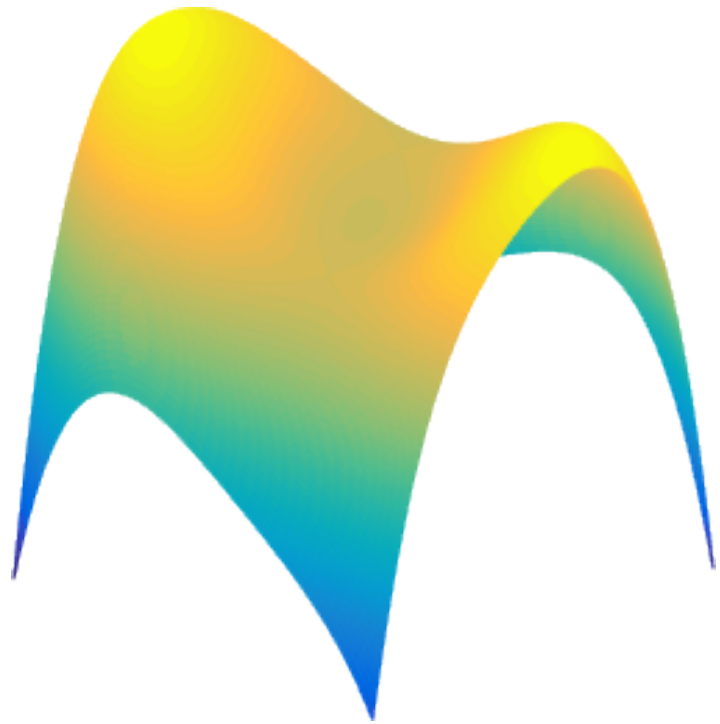
- Part IV: Robustness to Corruptions and Ill-Conditioning

  median truncation, least absolute deviation, scaled gradient descent

Nonconvex estimation problems are everywhere

Empirical risk minimization is usually nonconvex

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}; \text{data}) \quad \rightarrow \quad \text{loss function may be nonconvex}$$

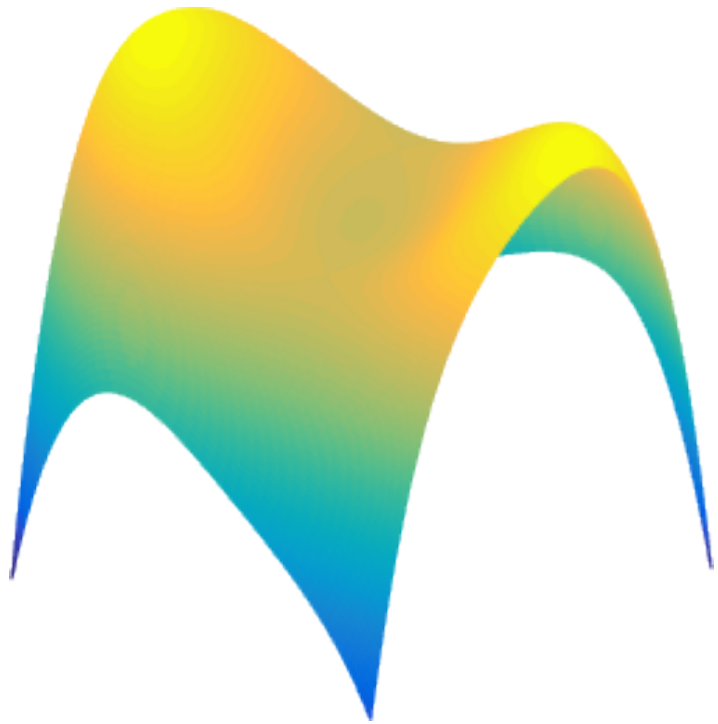


Nonconvex estimation problems are everywhere

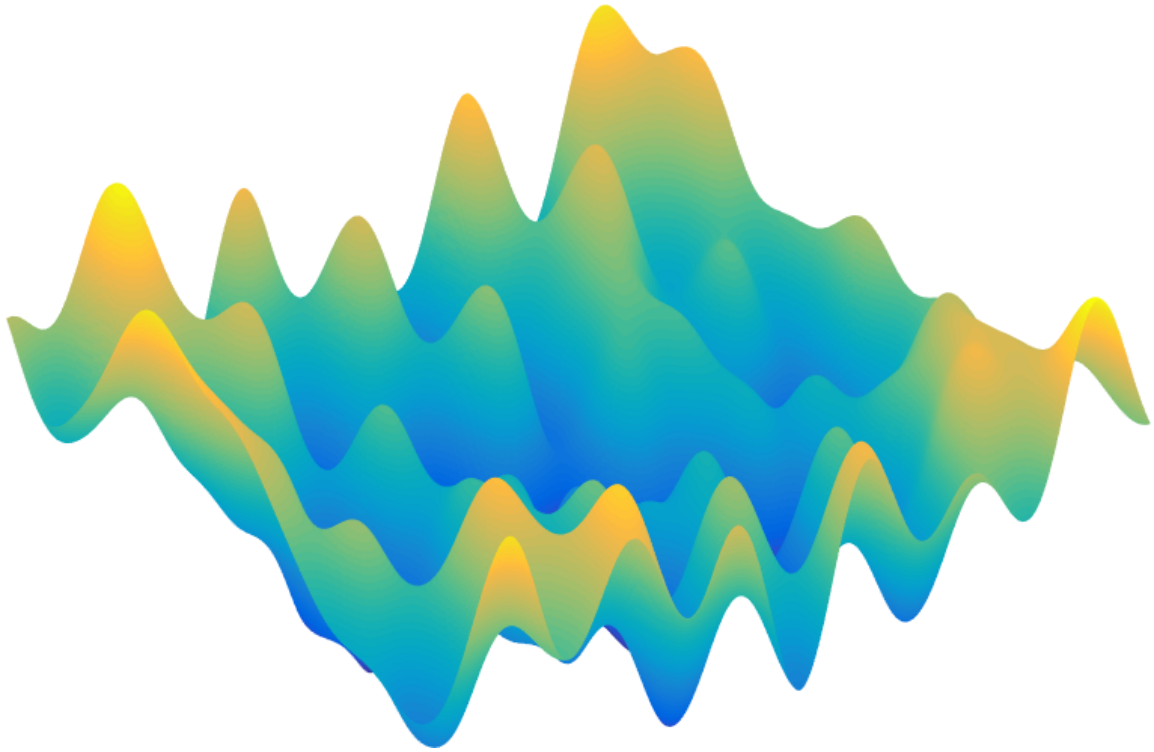
Empirical risk minimization is usually nonconvex

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}; \text{data}) \quad \rightarrow \quad \text{loss function may be nonconvex}$$

- low-rank matrix completion
- blind deconvolution
- dictionary learning
- mixture models
- deep learning
- ...



Nonconvex optimization may be super scary



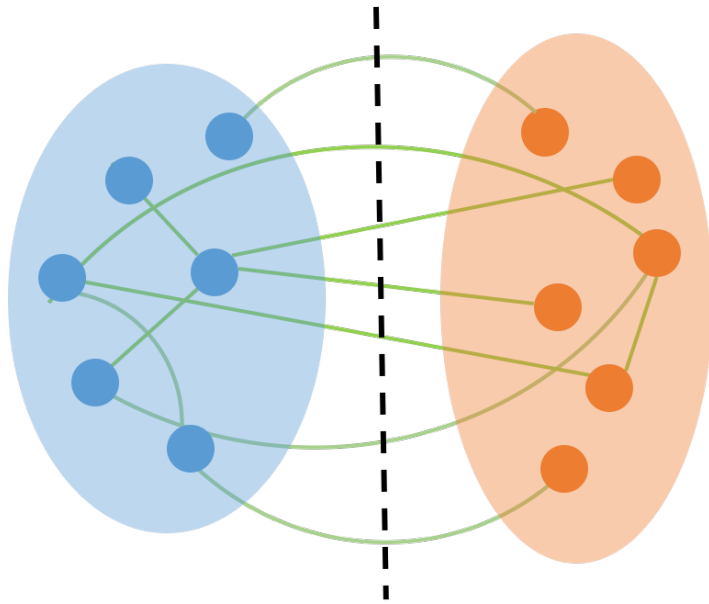
There may be bumps everywhere and exponentially many local optima

e.g. 1-layer neural net (Auer, Herbster, Warmuth '96; Vu '98)

Example: solving quadratic programs is hard

Finding maximum cut in a graph is about solving a quadratic program

$$\begin{aligned} \text{maximize}_{\boldsymbol{x}} \quad & \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x} \\ \text{subj. to} \quad & x_i^2 = 1, \quad i = 1, \cdots, n \end{aligned}$$



Example: solving quadratic programs is hard

"I can't find an efficient algorithm, but neither can all these people."

figure credit: coding horror

$1,000,000 question

One strategy: convex relaxation

Can relax into convex problems by

- finding convex surrogates (e.g. matrix completion)
- lifting into higher dimensions (e.g. Max-Cut)

Example of convex surrogate: matrix completion

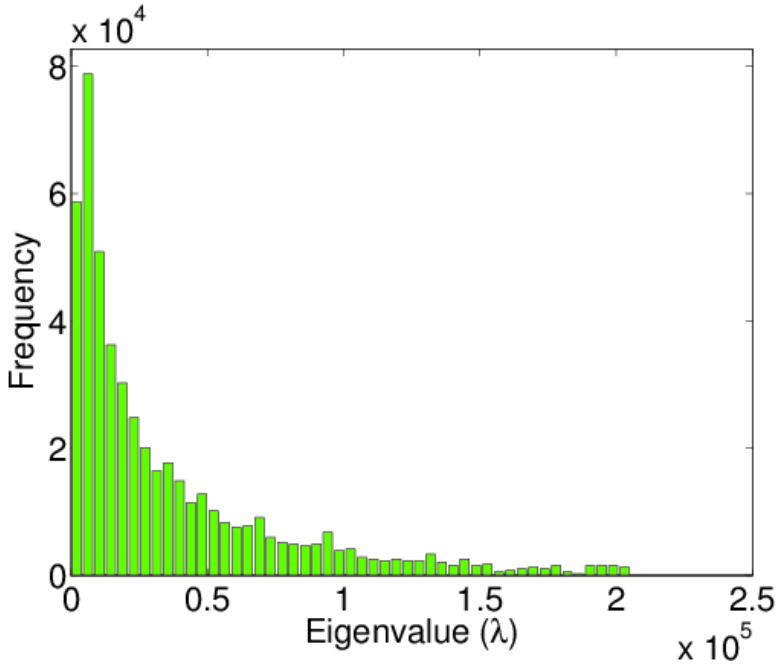
figure credit: Candès et al.

Netflix challenge

Predict unseen ratings

Low-rank modeling

figure credit: E. Candès



A few factors explain most of the data

Low-rank modeling

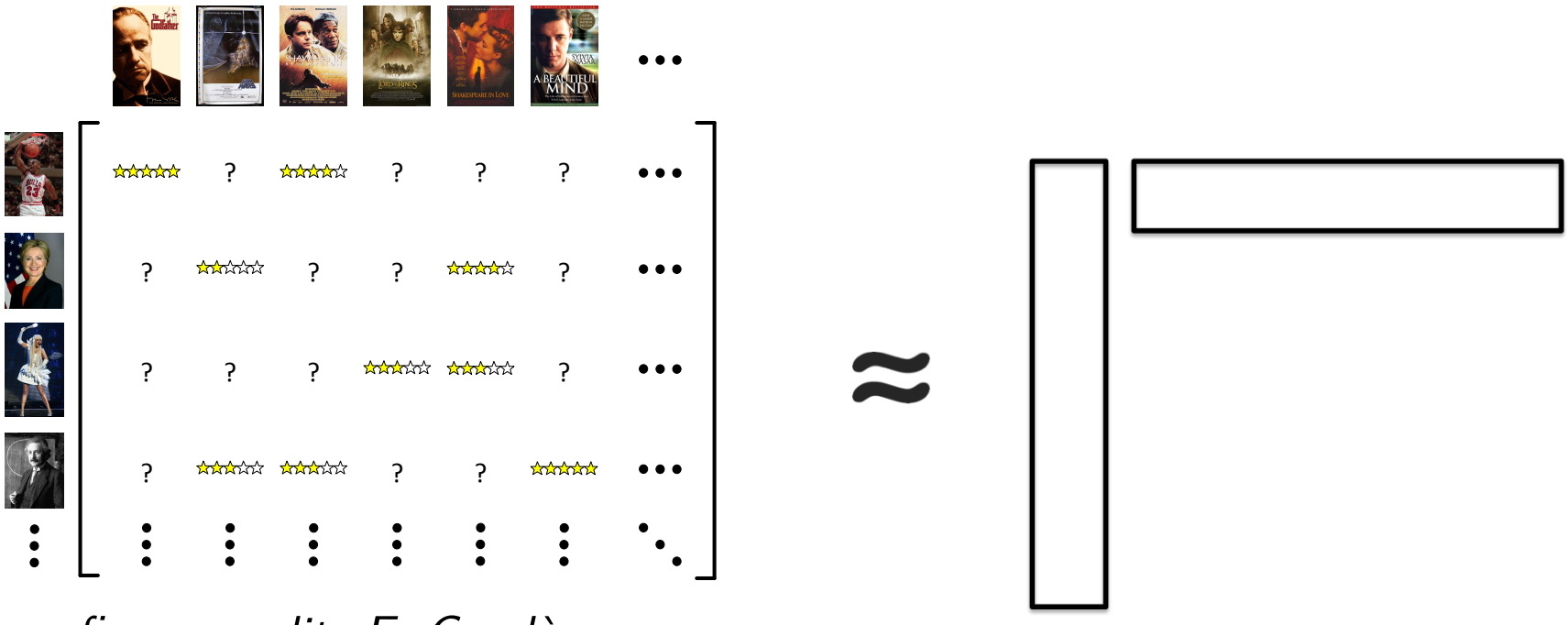


figure credit: E. Candès

A few factors explain most of the data $\longrightarrow$ low-rank approximation

How to exploit (approx.) low-rank structure in prediction?

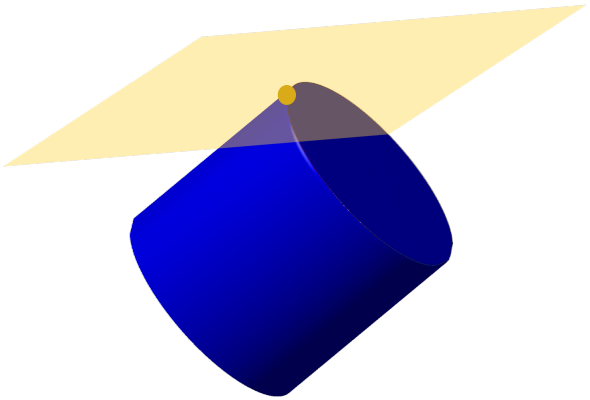
Example of convex surrogate: matrix completion

— Fazel '02, Recht, Parrilo, Fazel '10, Candès, Recht '09

minimize$_M$ rank(M) subj. to data constraints

⇩ cvx surrogate

minimize$_M$ nuc-norm(M) subj. to data constraints



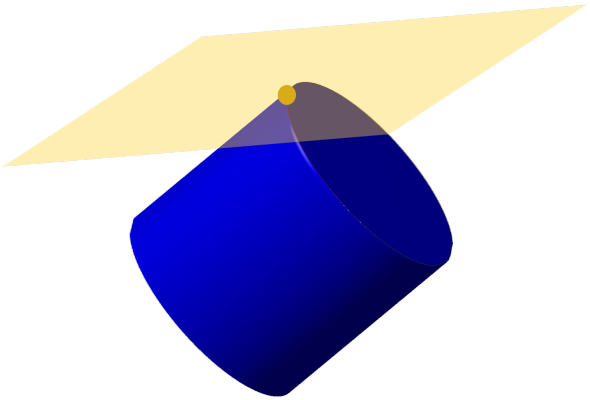
Example of convex surrogate: matrix completion

— Fazel '02, Recht, Parrilo, Fazel '10, Candès, Recht '09

minimize$_{\boldsymbol{M}}$ rank$(\boldsymbol{M})$ subj. to data constraints

⇩ cvx surrogate

minimize$_{\boldsymbol{M}}$ nuc-norm$(\boldsymbol{M})$ subj. to data constraints



robust variation used by Netflix

— Candès, Li, Ma, Wright '10

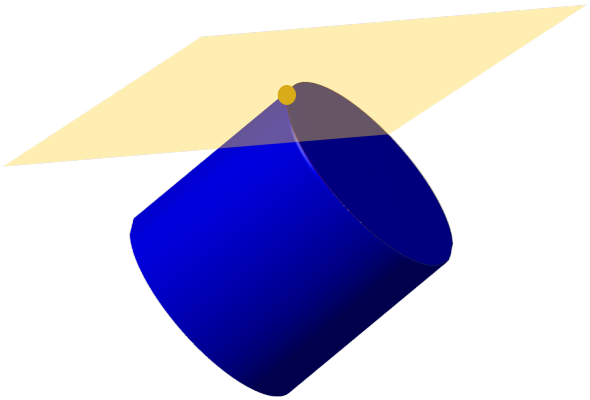
Example of convex surrogate: matrix completion

— Fazel '02, Recht, Parrilo, Fazel '10, Candès, Recht '09

minimize$_{\boldsymbol{M}}$ rank$(\boldsymbol{M})$ subj. to data constraints

⇩ cvx surrogate

minimize$_{\boldsymbol{M}}$ nuc-norm$(\boldsymbol{M})$ subj. to data constraints



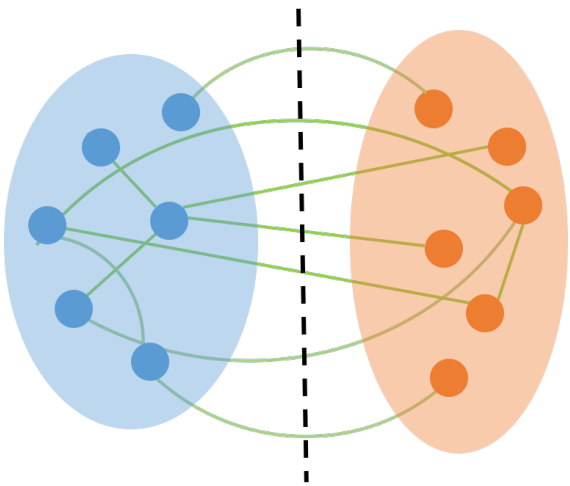
robust variation used by Netflix

— Candès, Li, Ma, Wright '10

Problem: operate in *full* matrix space even though $\boldsymbol{X}$ is low-rank

Example of lifting: Max-Cut

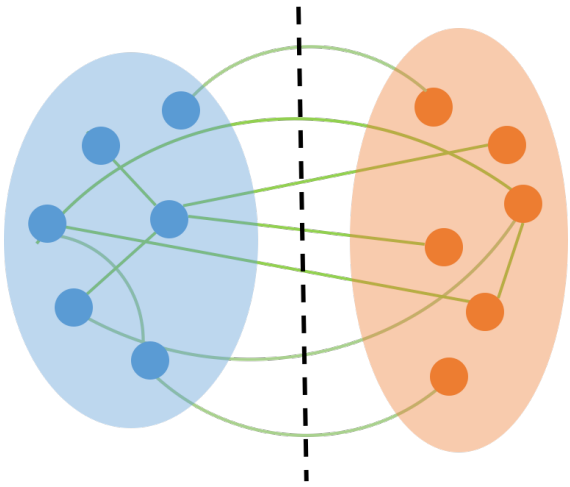
— Goemans, Williamson '95



$$\begin{aligned} \text{maximize}_{\boldsymbol{x}} \quad & \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x} \\ \text{subj. to} \quad & x_i^2 = 1, \quad i = 1, \cdots, n \end{aligned}$$

Example of lifting: Max-Cut

— Goemans, Williamson '95



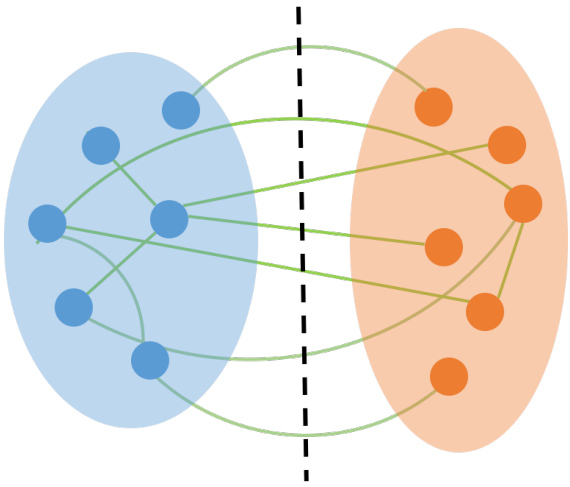
$$\begin{aligned} \text{maximize}_{\boldsymbol{x}} \quad & \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x} \\ \text{subj. to} \quad & x_i^2 = 1, \quad i = 1, \cdots, n \end{aligned}$$

⇩ let $\boldsymbol{X}$ be $\boldsymbol{x}\boldsymbol{x}^\top$

$$\begin{aligned} \text{maximize}_{\boldsymbol{X}} \quad & \langle \boldsymbol{X}, \boldsymbol{W} \rangle \\ \text{subj. to} \quad & \boldsymbol{X}_{i,i} = 1, \quad i = 1, \cdots, n \\ & \boldsymbol{X} \succeq \boldsymbol{0} \\ & \text{rank}(\boldsymbol{X}) = 1 \end{aligned}$$

Example of lifting: Max-Cut

— Goemans, Williamson '95



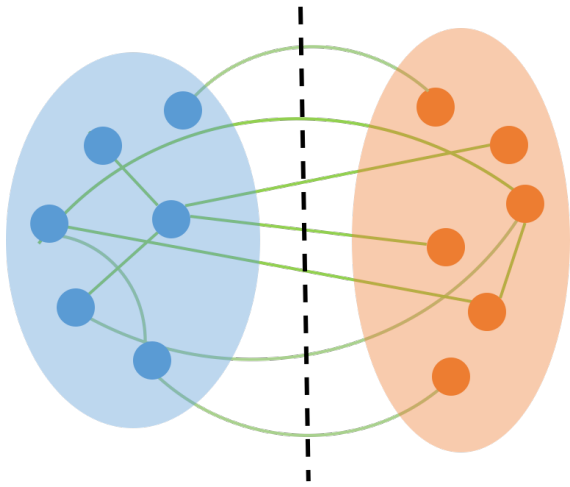
$$\begin{aligned} \text{maximize}_{\boldsymbol{x}} \quad & \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x} \\ \text{subj. to} \quad & x_i^2 = 1, \quad i = 1, \cdots, n \end{aligned}$$

⇩ let $\boldsymbol{X}$ be $\boldsymbol{x}\boldsymbol{x}^\top$

$$\begin{aligned} \text{maximize}_{\boldsymbol{X}} \quad & \langle \boldsymbol{X}, \boldsymbol{W} \rangle \\ \text{subj. to} \quad & \boldsymbol{X}_{i,i} = 1, \quad i = 1, \cdots, n \\ & \boldsymbol{X} \succeq \boldsymbol{0} \\ & \sout{\text{rank}(\boldsymbol{X}) = 1} \end{aligned}$$

Example of lifting: Max-Cut

— Goemans, Williamson '95



$$\begin{aligned} \text{maximize}_{\boldsymbol{x}} \quad & \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x} \\ \text{subj. to} \quad & x_i^2 = 1, \quad i = 1, \cdots, n \end{aligned}$$

⇩ let $\boldsymbol{X}$ be $\boldsymbol{x}\boldsymbol{x}^\top$

$$\begin{aligned} \text{maximize}_{\boldsymbol{X}} \quad & \langle \boldsymbol{X}, \boldsymbol{W} \rangle \\ \text{subj. to} \quad & \boldsymbol{X}_{i,i} = 1, \quad i = 1, \cdots, n \\ & \boldsymbol{X} \succeq \boldsymbol{0} \\ & \cancel{\text{rank}(\boldsymbol{X}) = 1} \end{aligned}$$

Problem: explosion in dimensions ($\mathbb{R}^n \to \mathbb{R}^{n \times n}$)

How about optimizing nonconvex problems directly without lifting?

Nonconvex optimization

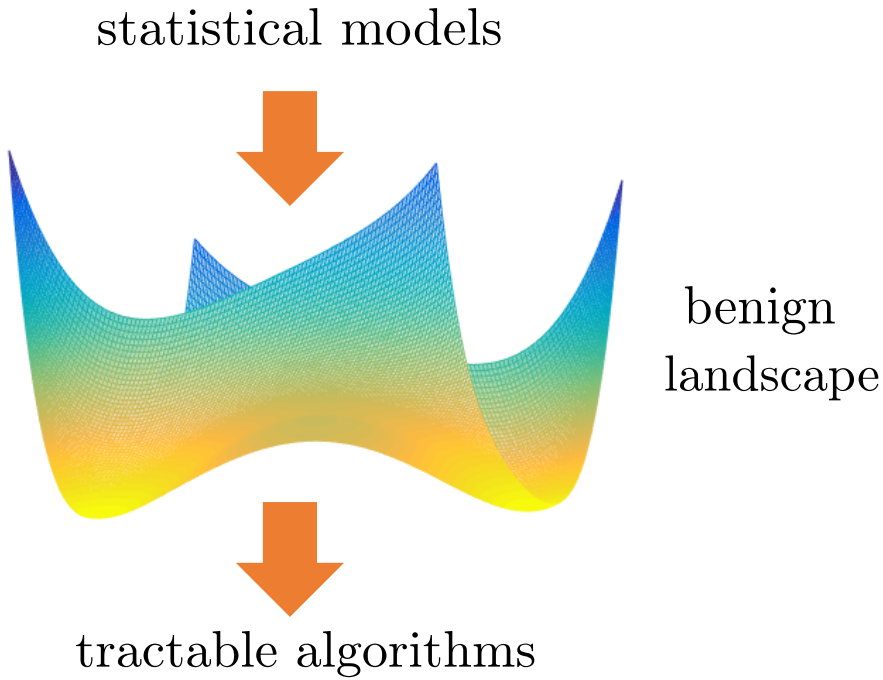
Complicated nonconvex problems are solved on a daily basis via simple algorithms such as stochastic gradient descent

Nonconvex optimization

Complicated nonconvex problems are solved on a daily basis via simple algorithms such as stochastic gradient descent

- How come simple nonconvex algorithms work so well in practice?

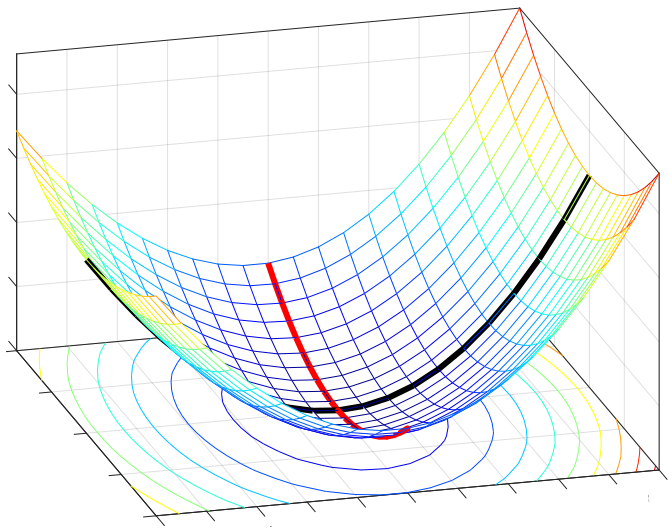
Statistical models come to rescue



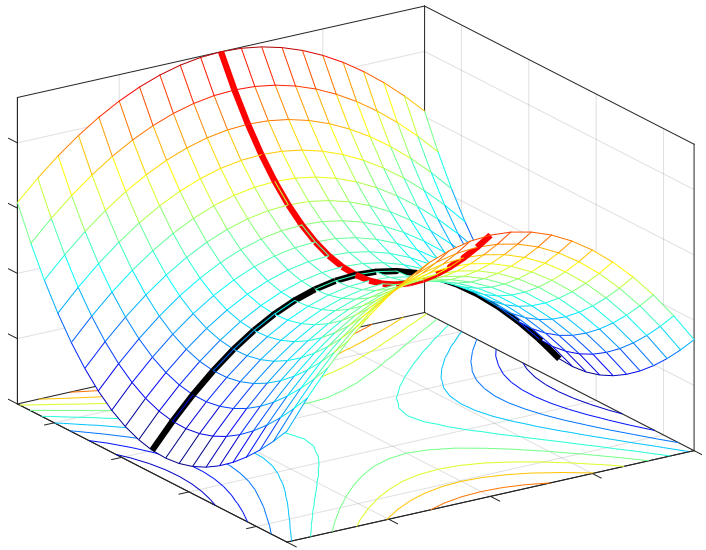
When data are generated by certain statistical models, problems are often much nicer than worst-case instances

Sometimes they are much nicer than we think

Under certain statistical models,
we see benign global geometry: no spurious local optima



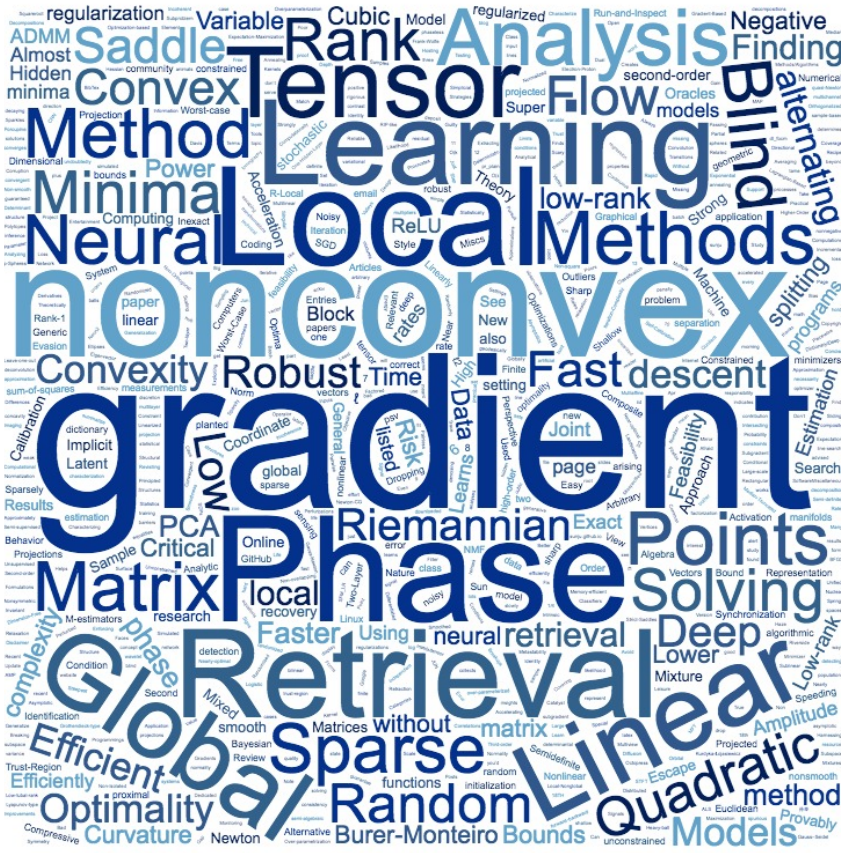
global minimum



saddle point

Even the simplest possible nonconvex methods
might be remarkably efficient under suitable statistical models

Nonconvex optimization with guarantees



Phase retrieval: Gerchberg-Saxton '72, Netrapalli et al. '13, Candès, Li, Soltanolkotabi '14, Chen, Candès '15, Cai, Li, Ma '15, Zhang et al. 16, Wang et al. '16, Sun et al. '16, Ma et al. '17, Chen et al. '18, ...

Matrix completion: Keshavan et al. '09, Jain et al. '09, Hardt '13, Sun, Luo '15, Chen, Wainwright '15, Zheng, Lafferty '16, Ge et al. '16, Jin et al. '16, Ma et al. '17, ...

Matrix sensing: Jain et al. '13, Tu et al. '15, Zheng, Lafferty '15, Bhojanapalli et al. 16, Li, Zhu, Tang '18, ...

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

Dictionary learning: Arora et al. '14, Sun et al. '15, Chatterji, Bartlett '17, ...

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

- *http://sunju.org/research/nonconvex/*
- *"Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview," Y. Chi, Y. M. Lu, and Y. Chen, IEEE Trans. on Signal Processing, vol. 67, no. 20, pp. 5239-5269, 2019.*

A bit preliminaries of optimization

Unconstrained optimization

Consider an unconstrained optimization problem

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

Definition 1 (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 2 (second-order critical points)

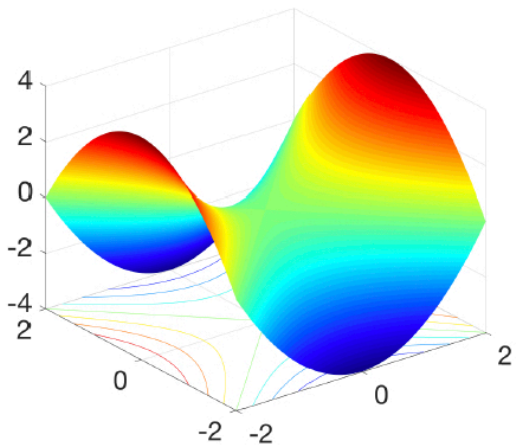
A second-order critical point $\boldsymbol{x}$ satisfies

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

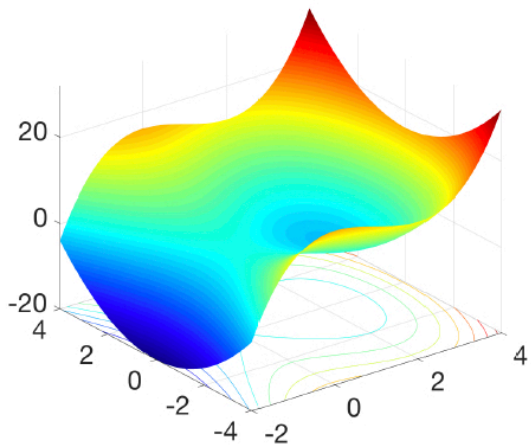
Several types of critical points

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

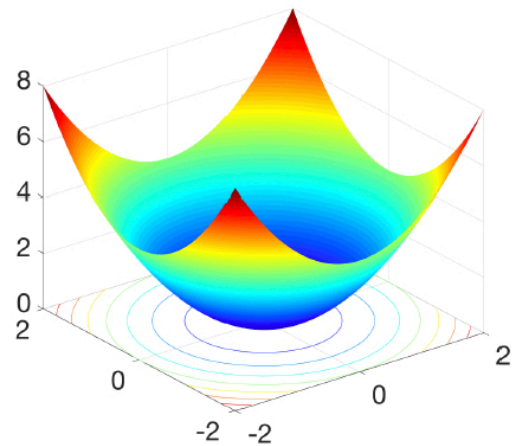
- $\nabla^2 f(\boldsymbol{x}) \prec \mathbf{0}$ $\quad\rightarrow\quad$ local maximum
- $\nabla^2 f(\boldsymbol{x}) \succ \mathbf{0}$ $\quad\rightarrow\quad$ local minimum
- $\lambda_{\min}(\nabla^2 f(\boldsymbol{x})) < 0$ $\quad\rightarrow\quad$ *strict* saddle point



(a) strict saddle



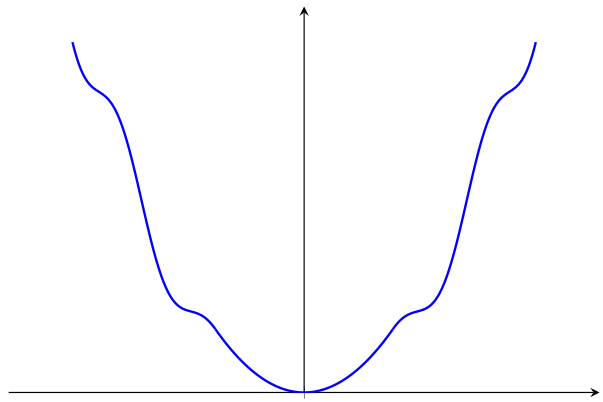
(b) local minimum



(c) global minimum

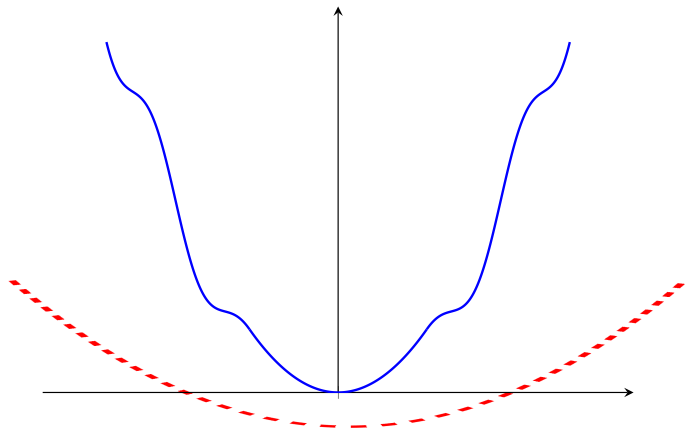
figure credit: Li et al. '16

Gradient descent theory



Two standard conditions that enable geometric convergence of GD

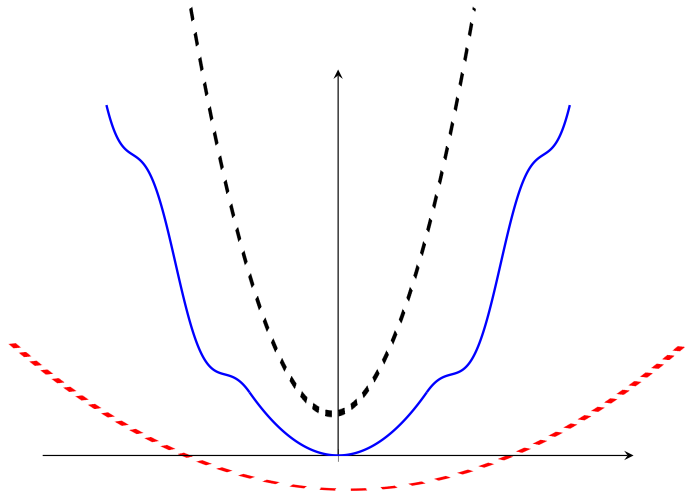
Gradient descent theory



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

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

- Condition number β/α determines rate of convergence

Gradient descent theory revisited

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

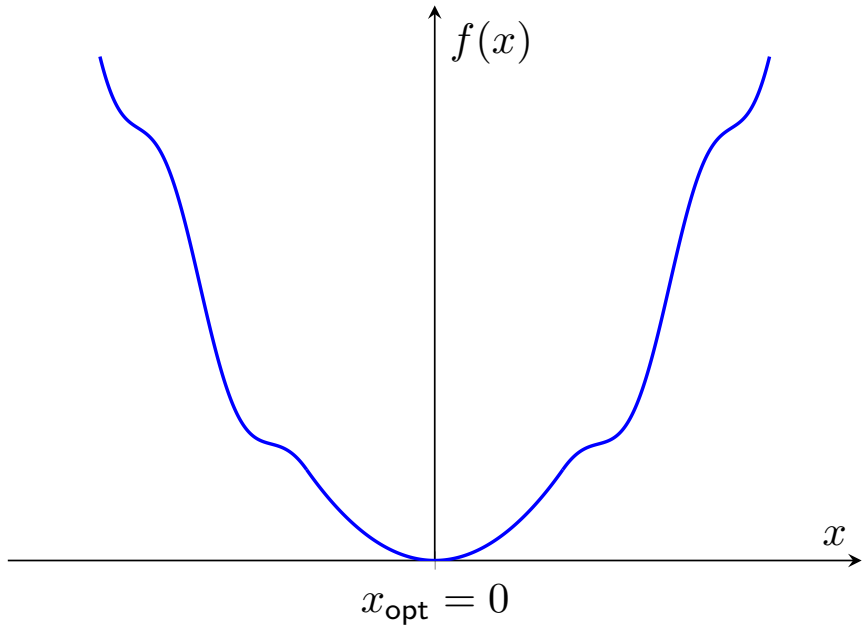
$$\mathbf{0} \preceq \alpha \boldsymbol{I} \preceq \nabla^2 f(\boldsymbol{x}) \preceq \beta \boldsymbol{I}, \qquad \forall \boldsymbol{x}$$

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

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

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

Regularity Condition (RC)



Definition 3 (Regularity Condition (RC))

$\boldsymbol{g}(\cdot)$ is said to obey RC(μ, λ, ζ) for some $\mu, \lambda, \zeta > 0$ if

$$2\langle \boldsymbol{g}(\boldsymbol{x}), \boldsymbol{x} - \boldsymbol{x}_{\text{opt}} \rangle \geq \mu \|\boldsymbol{g}(\boldsymbol{x})\|_2^2 + \lambda \|\boldsymbol{x} - \boldsymbol{x}_{\text{opt}}\|_2^2 \quad \forall \boldsymbol{x}$$

Convergence under RC

ℓ_2 error contraction: The update rule ($\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \boldsymbol{g}(\boldsymbol{x}^t)$) with $\eta = \mu$ obeys

$$\|\boldsymbol{x}^{t+1} - \boldsymbol{x}_{\mathsf{opt}}\|_2 \leq (1 - \mu\lambda) \|\boldsymbol{x}^t - \boldsymbol{x}_{\mathsf{opt}}\|_2$$

- $\boldsymbol{g}(\cdot)$: more general search directions
  - example: in vanilla GD, $\boldsymbol{g}(\boldsymbol{x}) = \nabla f(\boldsymbol{x})$

Convergence under RC

ℓ_2 error contraction: The update rule ($\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \boldsymbol{g}(\boldsymbol{x}^t)$) with $\eta = \mu$ obeys

$$\|\boldsymbol{x}^{t+1} - \boldsymbol{x}_{\text{opt}}\|_2 \leq (1 - \mu\lambda) \|\boldsymbol{x}^t - \boldsymbol{x}_{\text{opt}}\|_2$$

- $\boldsymbol{g}(\cdot)$: more general search directions
  - example: in vanilla GD, $\boldsymbol{g}(\boldsymbol{x}) = \nabla f(\boldsymbol{x})$
- The product $\mu\lambda$ determines the rate of convergence

Convergence under RC

ℓ_2 error contraction: The update rule ($\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \boldsymbol{g}(\boldsymbol{x}^t)$) with $\eta = \mu$ obeys

$$\|\boldsymbol{x}^{t+1} - \boldsymbol{x}_{\text{opt}}\|_2 \leq (1 - \mu\lambda) \|\boldsymbol{x}^t - \boldsymbol{x}_{\text{opt}}\|_2$$

- $\boldsymbol{g}(\cdot)$: more general search directions
  - example: in vanilla GD, $\boldsymbol{g}(\boldsymbol{x}) = \nabla f(\boldsymbol{x})$
- The product $\mu\lambda$ determines the rate of convergence
- Attains ε-accuracy within $O(\frac{1}{\mu\lambda} \log \frac{1}{\varepsilon})$ iterations

RC = one-point strong convexity + smoothness

- One-point α-strong convexity:

$$f(\boldsymbol{x}_{\mathsf{opt}}) - f(\boldsymbol{x}) \geq \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}_{\mathsf{opt}} - \boldsymbol{x} \rangle + \frac{\alpha}{2} \|\boldsymbol{x} - \boldsymbol{x}_{\mathsf{opt}}\|_2^2 \tag{1}$$

- β-smoothness:

$$\begin{aligned} f(\boldsymbol{x}_{\mathsf{opt}}) - f(\boldsymbol{x}) &\leq f\Big(\boldsymbol{x} - \frac{1}{\beta}\nabla f(\boldsymbol{x})\Big) - f(\boldsymbol{x}) \\ &\leq \Big\langle \nabla f(\boldsymbol{x}), -\frac{1}{\beta}\nabla f(\boldsymbol{x}) \Big\rangle + \frac{\beta}{2}\Big\|\frac{1}{\beta}\nabla f(\boldsymbol{x})\Big\|_2^2 \\ &= -\frac{1}{2\beta}\|\nabla f(\boldsymbol{x})\|_2^2 \end{aligned} \tag{2}$$

RC = one-point strong convexity + smoothness

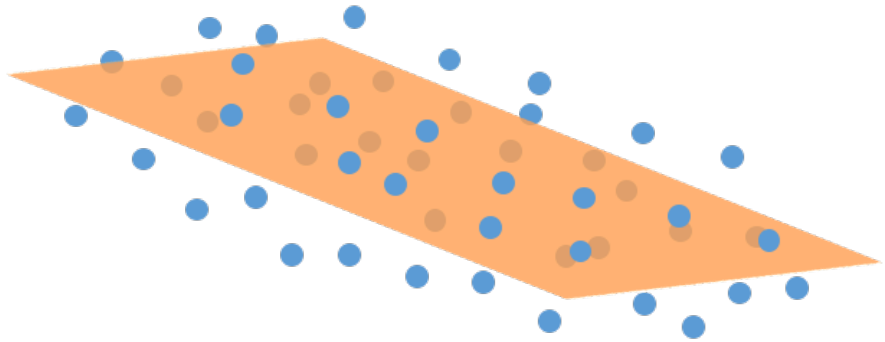
Combining (1) and (2) yields

$$\langle \nabla f(\boldsymbol{x}), \boldsymbol{x} - \boldsymbol{x}_{\mathsf{opt}} \rangle \geq \frac{\alpha}{2} \|\boldsymbol{x} - \boldsymbol{x}_{\mathsf{opt}}\|_2^2 + \frac{1}{2\beta} \|\nabla f(\boldsymbol{x})\|_2^2 \tag{3}$$

— *RC holds with* $\mu = 1/\beta$ *and* $\lambda = \alpha$

A toy example: rank-1 matrix factorization

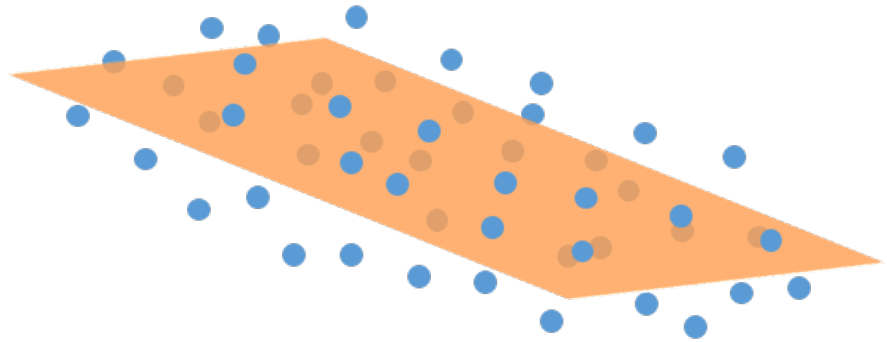
Revisiting PCA



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

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

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*

Optimization viewpoint

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:

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

To simplify exposition, set $r = 1$:

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4} \|\boldsymbol{x}\boldsymbol{x}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2$$

Questions

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4}\|\boldsymbol{x}\boldsymbol{x}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2$$

- Where / what are the critical points?
- What does the curvature behave like, at least locally around the global minimizer?

Critical points of $f(\cdot)$

$\boldsymbol{x}$ is a critical point, i.e. $\nabla f(\boldsymbol{x}) = (\boldsymbol{x}\boldsymbol{x}^\top - \boldsymbol{M})\boldsymbol{x} = \boldsymbol{0}$

$$\Updownarrow$$

$$\boldsymbol{M}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2\boldsymbol{x}$$

$$\Updownarrow$$

$\boldsymbol{x}$ aligns with an eigenvector of $\boldsymbol{M}$ or $\boldsymbol{x} = \boldsymbol{0}$

Since $\boldsymbol{M}\boldsymbol{u}_i = \lambda_i\boldsymbol{u}_i$, the set of critical points is given by

$$\{\boldsymbol{0}\} \cup \{\pm\sqrt{\lambda_i}\boldsymbol{u}_i,\ i = 1, \ldots, n\}$$

Categorization of critical points

The critical points can be further categorized based on the **Hessians**:

$$\nabla^2 f(\boldsymbol{x}) := 2\boldsymbol{x}\boldsymbol{x}^\top + \|\boldsymbol{x}\|_2^2 \boldsymbol{I} - \boldsymbol{M}$$

- For any non-zero critical point $\boldsymbol{x}_k = \pm\sqrt{\lambda_k}\boldsymbol{u}_k$:

$$\begin{aligned}
\nabla^2 f(\boldsymbol{x}_k) &= 2\lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^\top + \lambda_k \boldsymbol{I} - \boldsymbol{M} \\
&= 2\sigma_k \boldsymbol{u}_k \boldsymbol{u}_k^\top + \lambda_k \left(\sum_{i=1}^{n} \boldsymbol{u}_i \boldsymbol{u}_i^\top \right) - \sum_{i=1}^{n} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^\top \\
&= \sum_{i:i\neq k} (\lambda_k - \lambda_i) \boldsymbol{u}_i \boldsymbol{u}_i^\top + 2\lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^\top
\end{aligned}$$

Categorization of critical points

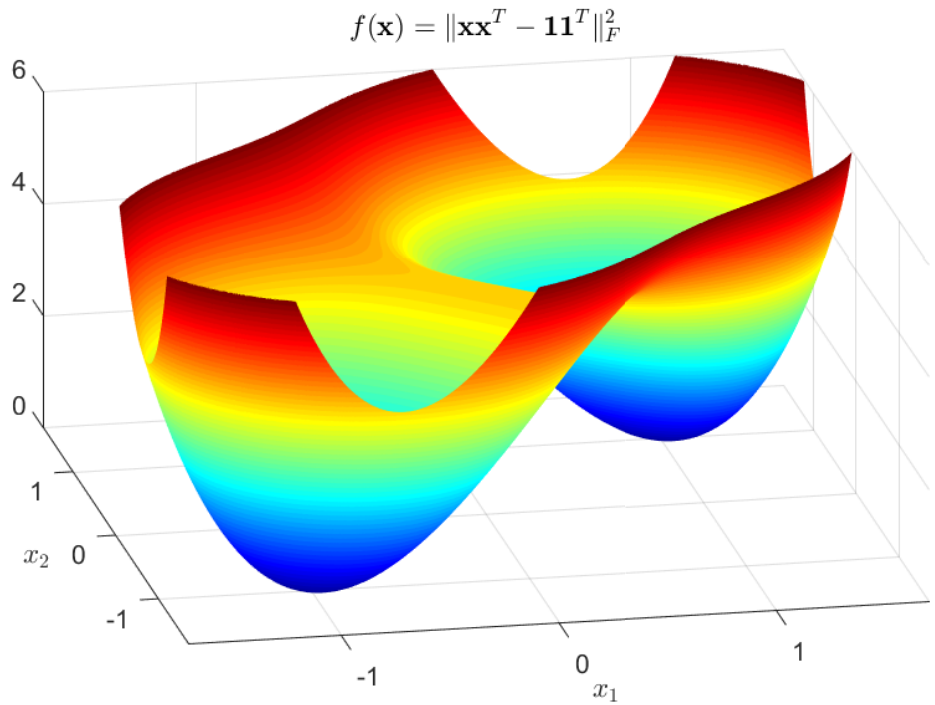
The critical points can be further categorized based on the **Hessians**:

$$\nabla^2 f(\boldsymbol{x}) := 2\boldsymbol{x}\boldsymbol{x}^\top + \|\boldsymbol{x}\|_2^2 \boldsymbol{I} - \boldsymbol{M}$$

- If $\lambda_1 > \lambda_2 \geq \ldots \geq \lambda_n \geq 0$, then
  - $\nabla^2 f(\boldsymbol{x}_1) \succ \boldsymbol{0} \quad \rightarrow$ local minima
  - $1 < k \leq n$: $\lambda_{\min}(\nabla^2 f(\boldsymbol{x}_k)) < 0$, $\lambda_{\max}(\nabla^2 f(\boldsymbol{x}_k)) > 0$ $\rightarrow$ strict saddle
  - $\boldsymbol{x} = \boldsymbol{0}$: $\nabla^2 f(\boldsymbol{0}) \preceq \boldsymbol{0} \quad \rightarrow$ local maxima

Good news: benign landscape

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

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

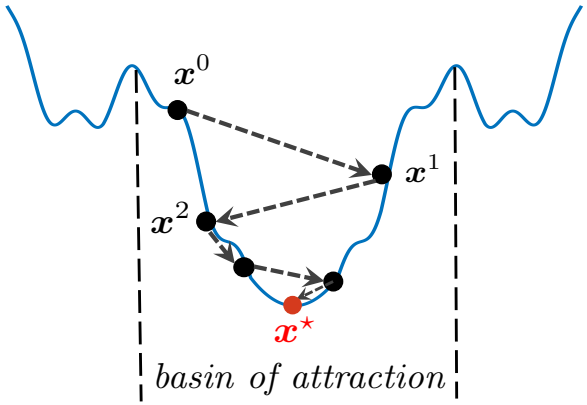
ℓ_2 **error contraction:** The GD iterates obey

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

as long as $\|\boldsymbol{x}^0 - \sqrt{\lambda_1}\boldsymbol{u}_1\|_2 \le \frac{\lambda_1 - \lambda_2}{15\sqrt{\lambda_1}}$

Two vignettes

Two-stage approach:



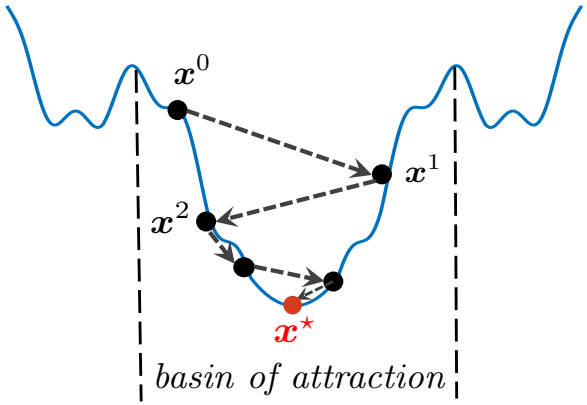
smart initialization

$+$

local refinement

Two vignettes

Two-stage approach:

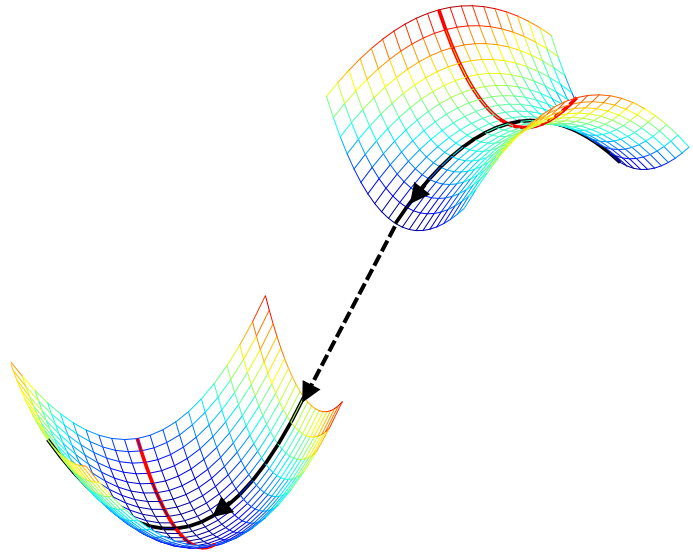


smart initialization

$+$

local refinement

Global landscape:



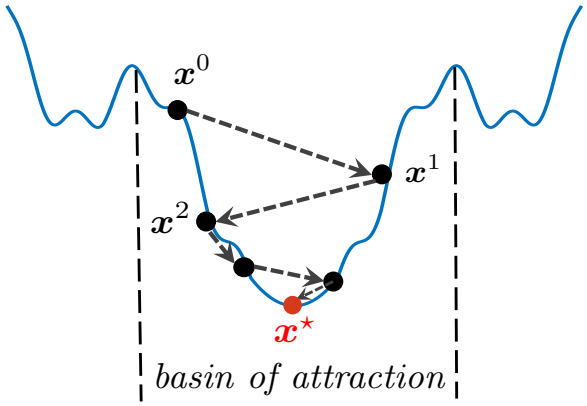
benign landscape

$+$

saddle-point escaping

Two vignettes

Two-stage approach:

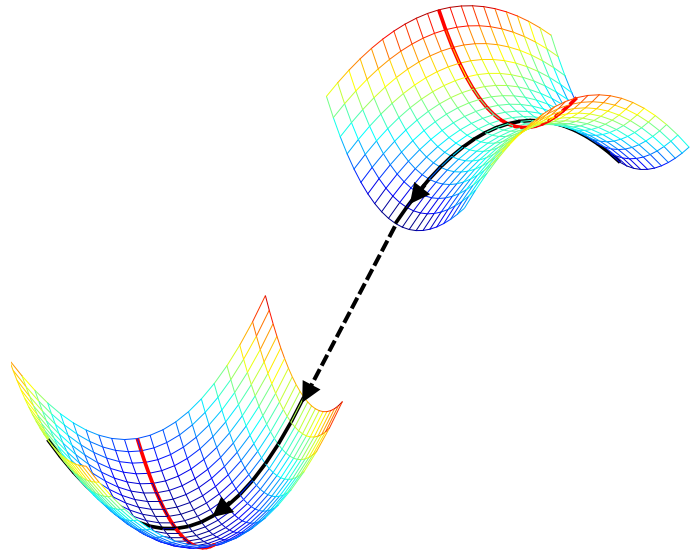


smart initialization

$+$

local refinement

Global landscape:



benign landscape

$+$

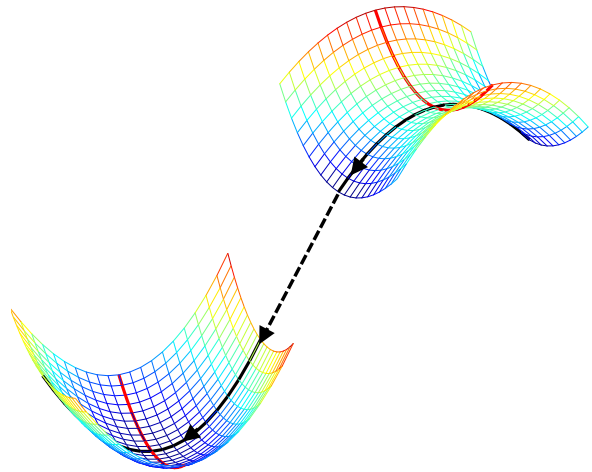
saddle-point escaping

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

Global landscape

Benign landscape:

- all local minima = global minima
- other critical points = strict saddle points

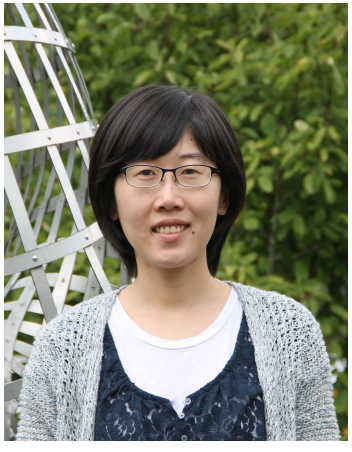


Saddle-point escaping algorithms:

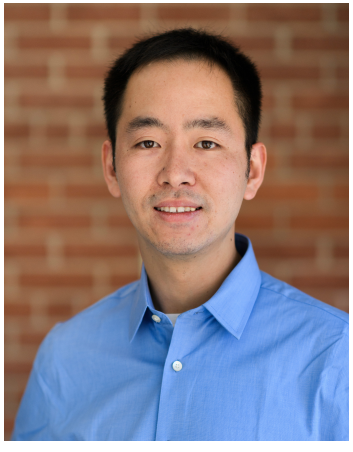
- trust-region methods;
- perturbed gradient descent;
- perturbed SGD;
- etc...

Check the recent overview: *Zhang, Qu, Wright "From Symmetry to Geometry: Tractable Nonconvex Problems"*

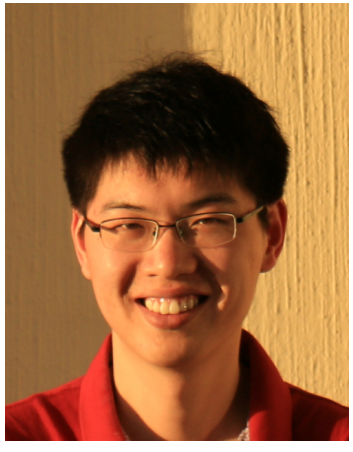
Nonconvex Optimization for High-Dimensional Signal Estimation: Spectral and Iterative Methods – Part II



Yuejie Chi
Carnegie Mellon



Yuxin Chen
Princeton



Cong Ma
UC Berkeley

EUSIPCO Tutorial, December 2020

Outline

- Part I: Introduction and Warm-Up

  Why nonconvex? basic concepts and a warm-up example (PCA)

- Part II: Gradient Descent and Implicit Regularization

  phase retrieval, matrix completion, random initialization

- Part III: Spectral Methods

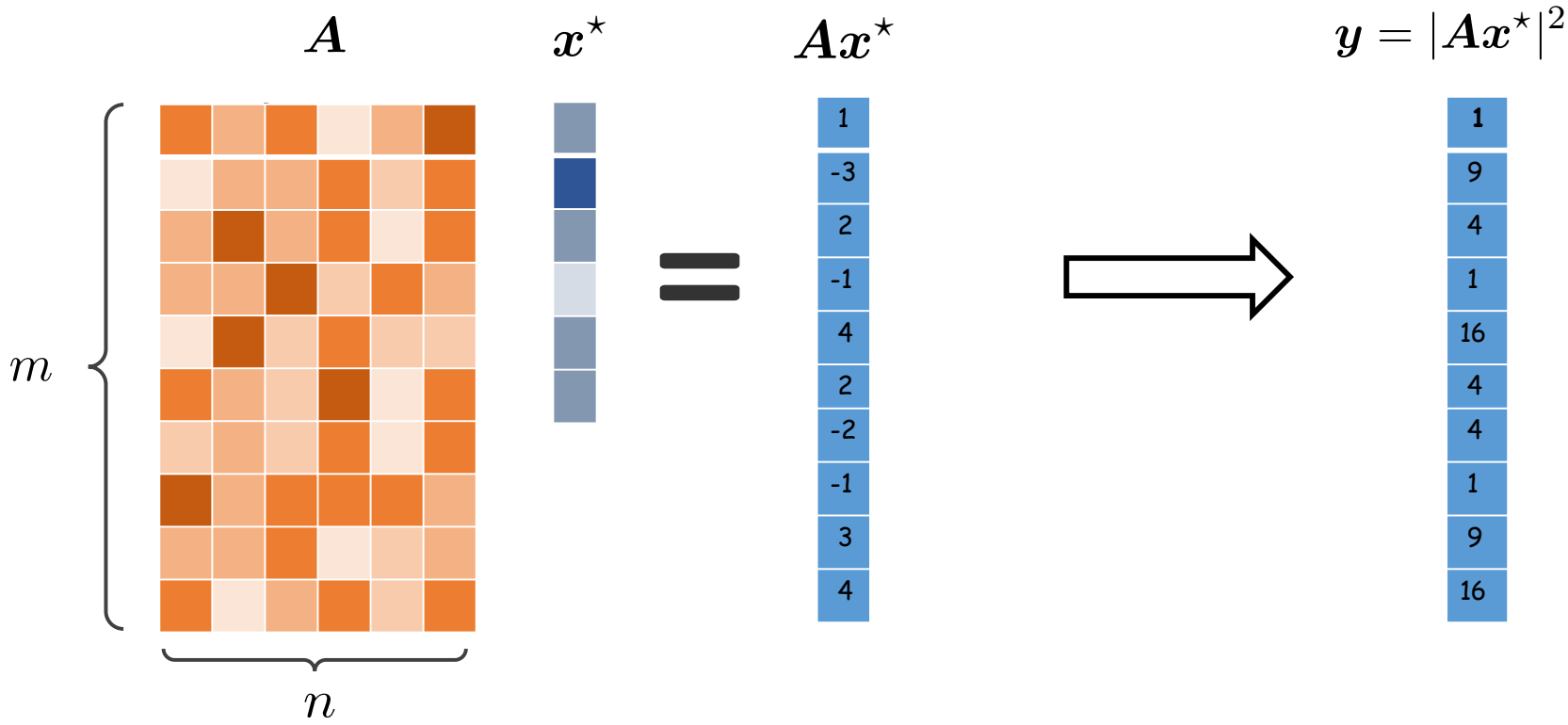
  a general recipe, ℓ_2 and ℓ_∞ guarantees, community detection

- Part IV: Robustness to Corruptions and Ill-Conditioning

  median truncation, least absolute deviation, scaled gradient descent

A case study: solving quadratic systems of equations

Solving quadratic systems of equations



Recover $\boldsymbol{x}^\star \in \mathbb{R}^n$ from m random quadratic measurements

$$y_k \quad = \quad (\boldsymbol{a}_k^\top \boldsymbol{x}^\star)^2, \qquad k = 1, \ldots, m$$

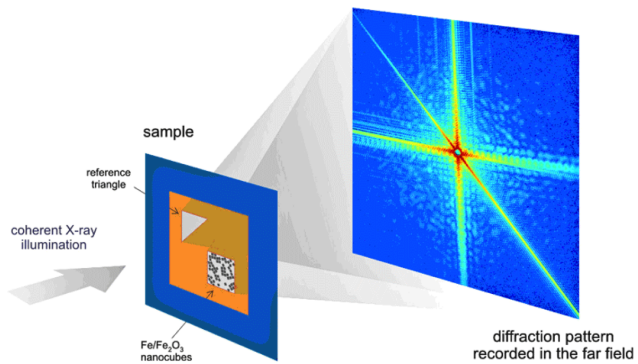
assume w.l.o.g. $\|\boldsymbol{x}^\star\|_2 = 1$

Motivation: phase retrieval

Detectors record intensities of diffracted rays

- electric field $x(t_1, t_2) \longrightarrow$ Fourier transform $\widehat{x}(f_1, f_2)$

figure credit: Stanford SLAC



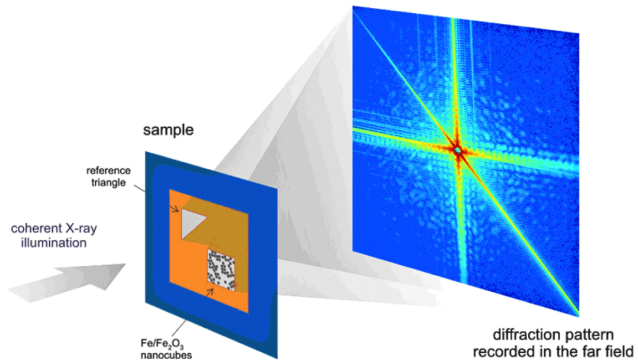
intensity of electrical field: $\left|\widehat{x}(f_1, f_2)\right|^2 = \left|\int x(t_1, t_2) e^{-i2\pi(f_1 t_1 + f_2 t_2)} \mathrm{d}t_1 \mathrm{d}t_2\right|^2$

Motivation: phase retrieval

Detectors record intensities of diffracted rays

- electric field $x(t_1, t_2) \longrightarrow$ Fourier transform $\widehat{x}(f_1, f_2)$

figure credit: Stanford SLAC

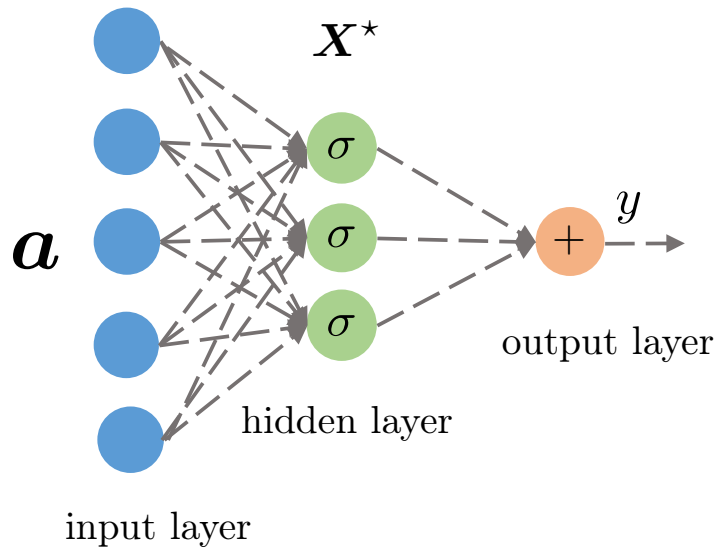


intensity of electrical field: $\left|\widehat{x}(f_1, f_2)\right|^2 = \left|\int x(t_1, t_2) e^{-i2\pi(f_1 t_1 + f_2 t_2)} \mathrm{d}t_1 \mathrm{d}t_2\right|^2$

Phase retrieval: recover signal $x(t_1, t_2)$ from intensity $\left|\widehat{x}(f_1, f_2)\right|^2$

Motivation: learning neural nets with quadratic activation

— *Soltanolkotabi, Javanmard, Lee '17, Li, Ma, Zhang '17*

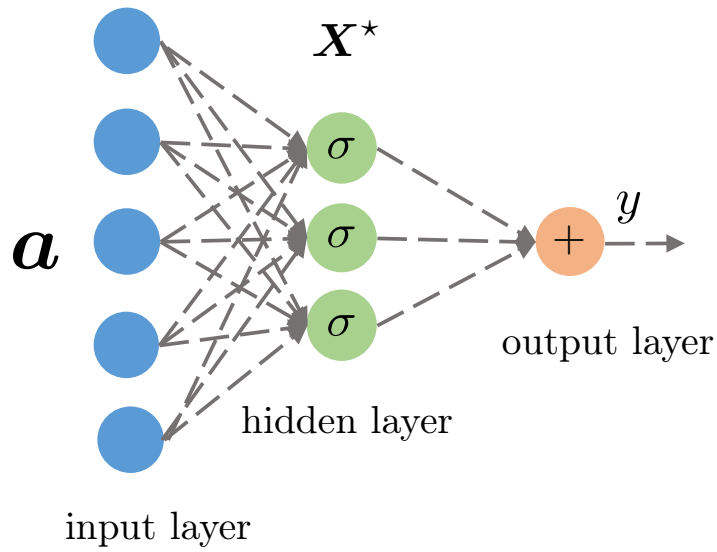


input features: $\boldsymbol{a}$; weights: $\boldsymbol{X}^\star = [\boldsymbol{x}_1^\star, \cdots, \boldsymbol{x}_r^\star]$

output: $$y = \sum_{i=1}^{r} \sigma(\boldsymbol{a}^\top \boldsymbol{x}_i^\star)$$

Motivation: learning neural nets with quadratic activation

— *Soltanolkotabi, Javanmard, Lee '17, Li, Ma, Zhang '17*

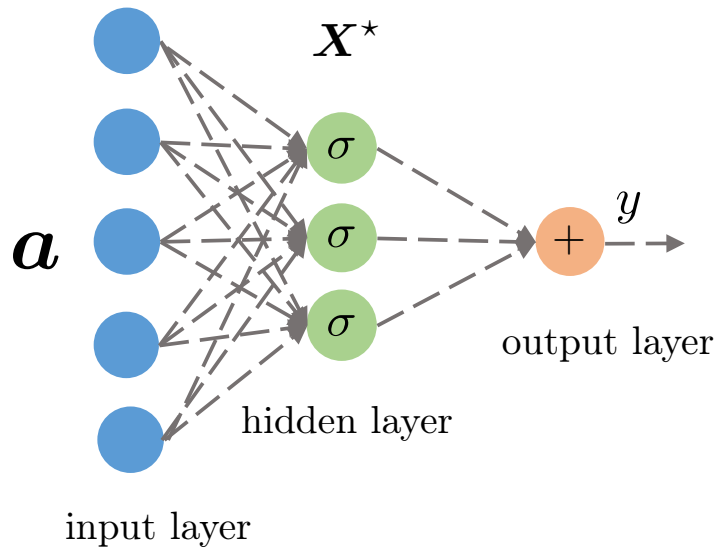


input features: $\boldsymbol{a}$; weights: $\boldsymbol{X}^\star = [\boldsymbol{x}_1^\star, \cdots, \boldsymbol{x}_r^\star]$

$$\text{output:} \quad y = \sum_{i=1}^{r} \sigma(\boldsymbol{a}^\top \boldsymbol{x}_i^\star) \overset{\sigma(z)=z^2}{:=} \sum_{i=1}^{r} (\boldsymbol{a}^\top \boldsymbol{x}_i^\star)^2$$

Motivation: learning neural nets with quadratic activation

— *Soltanolkotabi, Javanmard, Lee '17, Li, Ma, Zhang '17*



input features: $\boldsymbol{a}$; weights: $\boldsymbol{X}^\star = [\boldsymbol{x}_1^\star, \cdots, \boldsymbol{x}_r^\star]$

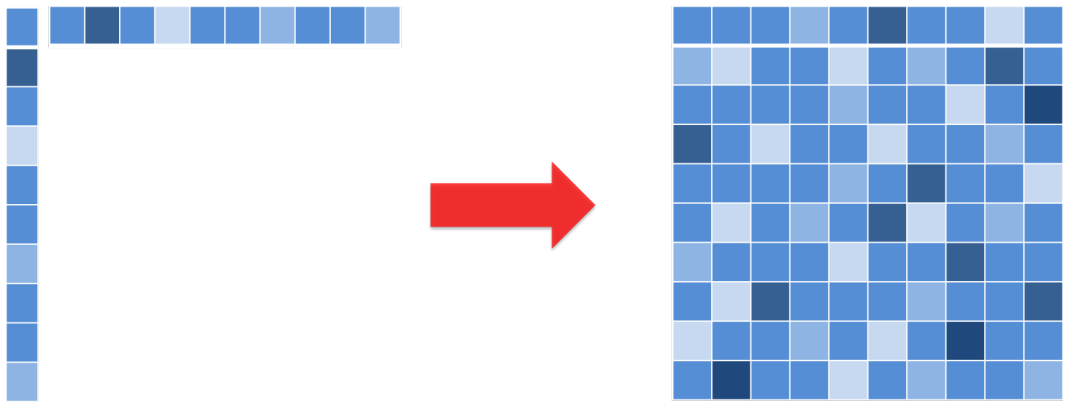
$$\text{output:} \quad y = \sum_{i=1}^{r} \sigma(\boldsymbol{a}^\top \boldsymbol{x}_i^\star) \overset{\sigma(z)=z^2}{:=} \sum_{i=1}^{r} (\boldsymbol{a}^\top \boldsymbol{x}_i^\star)^2$$

We consider simplest model when $r = 1$ (higher r is similar)

An equivalent view: low-rank factorization

Introduce $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^\top$ to linearize constraints

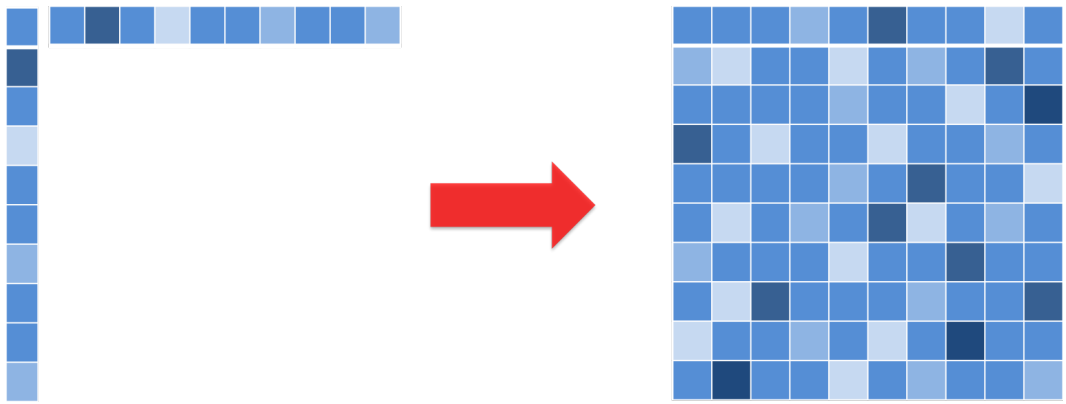
$$y_k = (\boldsymbol{a}_k^\top \boldsymbol{x})^2 = \boldsymbol{a}_k^\top (\boldsymbol{x}\boldsymbol{x}^\top)\boldsymbol{a} \quad \Longrightarrow \quad y_k = \boldsymbol{a}_k^\top \boldsymbol{X} \boldsymbol{a}_k$$



An equivalent view: low-rank factorization

Introduce $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^\top$ to linearize constraints

$$y_k = (\boldsymbol{a}_k^\top \boldsymbol{x})^2 = \boldsymbol{a}_k^\top (\boldsymbol{x}\boldsymbol{x}^\top)\boldsymbol{a} \qquad \Longrightarrow \qquad y_k = \boldsymbol{a}_k^\top \boldsymbol{X} \boldsymbol{a}_k$$

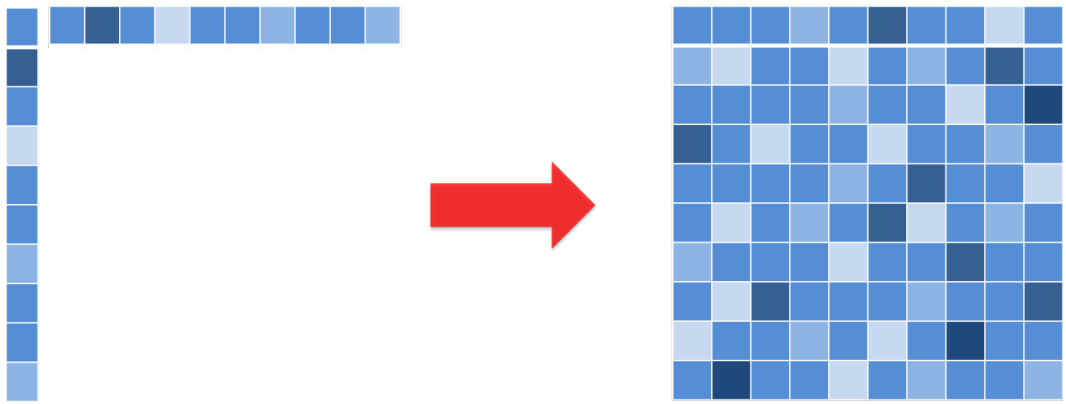


$$\begin{aligned} \text{find} \quad & \boldsymbol{X} \\ \text{s.t.} \quad & y_k = \boldsymbol{a}_k^\top \boldsymbol{X} \boldsymbol{a}_k, \qquad k = 1, \cdots, m \\ & \text{rank}(\boldsymbol{X}) = 1 \end{aligned}$$

An equivalent view: low-rank factorization

Introduce $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^\top$ to linearize constraints

$$y_k = (\boldsymbol{a}_k^\top \boldsymbol{x})^2 = \boldsymbol{a}_k^\top (\boldsymbol{x}\boldsymbol{x}^\top)\boldsymbol{a} \qquad \Longrightarrow \qquad y_k = \boldsymbol{a}_k^\top \boldsymbol{X} \boldsymbol{a}_k$$



$$\begin{aligned} \text{find} \quad & \boldsymbol{X} \\ \text{s.t.} \quad & y_k = \boldsymbol{a}_k^\top \boldsymbol{X} \boldsymbol{a}_k, \qquad k = 1, \cdots, m \\ & \text{rank}(\boldsymbol{X}) = 1 \end{aligned}$$

Solving quadratic systems is essentially low-rank matrix completion

A natural least-squares formulation

$$\text{given:} \qquad y_k \;=\; (\boldsymbol{a}_k^\top \boldsymbol{x}^\star)^2, \quad 1 \le k \le m$$

$$\Downarrow$$

$$\text{minimize}_{\boldsymbol{x}\in\mathbb{R}^n} \quad f(\boldsymbol{x}) = \frac{1}{4m}\sum_{k=1}^{m}\left[\left(\boldsymbol{a}_k^\top \boldsymbol{x}\right)^2 - y_k\right]^2$$

A natural least-squares formulation

$$\text{given:} \qquad y_k \;=\; (\boldsymbol{a}_k^\top \boldsymbol{x}^\star)^2, \quad 1 \le k \le m$$

$$\Downarrow$$

$$\text{minimize}_{\boldsymbol{x}\in\mathbb{R}^n} \quad f(\boldsymbol{x}) = \frac{1}{4m}\sum_{k=1}^{m}\Big[\big(\boldsymbol{a}_k^\top \boldsymbol{x}\big)^2 - y_k\Big]^2$$

- **pros:** often exact as long as sample size is sufficiently large

A natural least-squares formulation

$$\text{given:} \qquad y_k \;=\; (\boldsymbol{a}_k^\top \boldsymbol{x}^\star)^2, \quad 1 \le k \le m$$

$$\Downarrow$$

$$\text{minimize}_{\boldsymbol{x}\in\mathbb{R}^n} \quad f(\boldsymbol{x}) = \frac{1}{4m}\sum_{k=1}^{m}\Big[\big(\boldsymbol{a}_k^\top \boldsymbol{x}\big)^2 - y_k\Big]^2$$

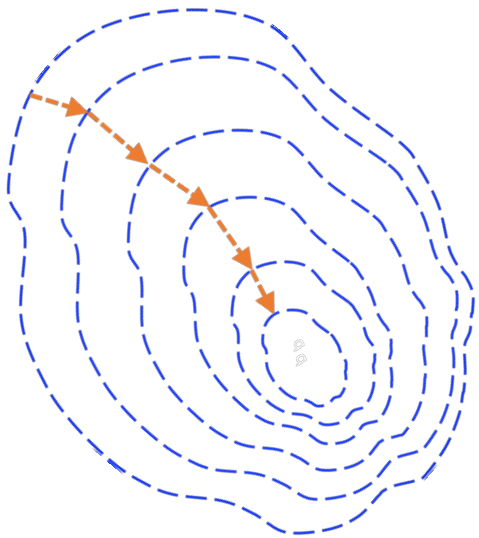
- **pros:** often exact as long as sample size is sufficiently large
- **cons:** $f(\cdot)$ is highly nonconvex
  $\longrightarrow$ *computationally challenging!*

Wirtinger flow (Candès, Li, Soltanolkotabi '14)

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4m} \sum_{k=1}^{m} \left[\left(\boldsymbol{a}_k^\top \boldsymbol{x} \right)^2 - y_k \right]^2$$

Wirtinger flow (Candès, Li, Soltanolkotabi '14)

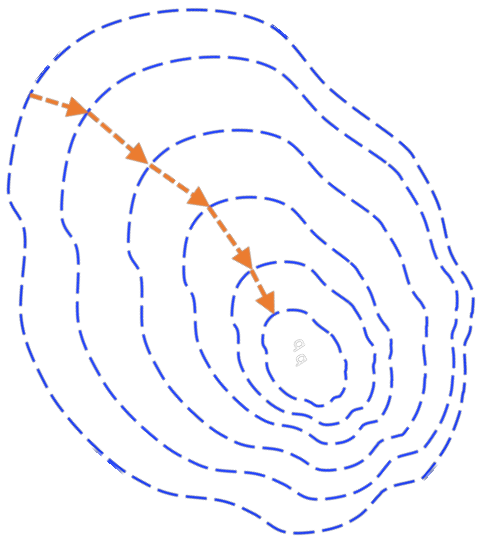
$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4m} \sum_{k=1}^{m} \left[\left(\boldsymbol{a}_k^\top \boldsymbol{x} \right)^2 - y_k \right]^2$$



- **spectral initialization:** $\boldsymbol{x}^0 \leftarrow$ leading eigenvector of certain data matrix

Wirtinger flow (Candès, Li, Soltanolkotabi '14)

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4m}\sum_{k=1}^{m}\left[\left(\boldsymbol{a}_k^\top \boldsymbol{x}\right)^2 - y_k\right]^2$$



- **spectral initialization:** $\boldsymbol{x}^0 \leftarrow$ leading eigenvector of certain data matrix
- **gradient descent:**

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta\,\nabla f(\boldsymbol{x}^t), \qquad t = 0, 1, \cdots$$

Spectral initialization

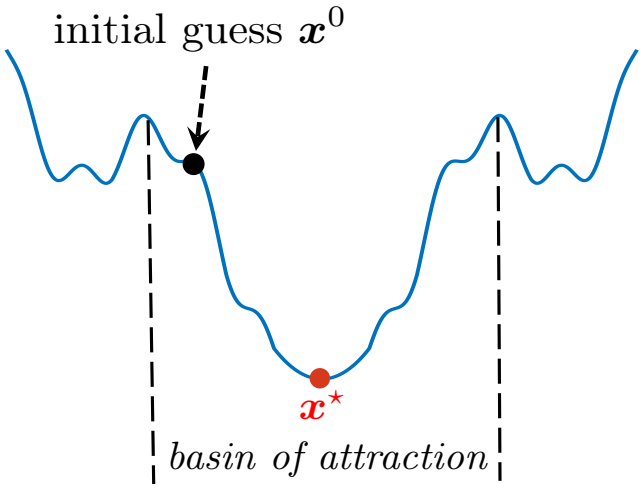
$\boldsymbol{x}^0 \longleftarrow$ leading eigenvector of

$$\boldsymbol{Y} := \frac{1}{m} \sum_{k=1}^{m} y_k \, \boldsymbol{a}_k \boldsymbol{a}_k^\top$$

Rationale: under random Gaussian design $\boldsymbol{a}_i \overset{\text{ind.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$,

$$\mathbb{E}[\boldsymbol{Y}] := \mathbb{E}\left[\frac{1}{m} \sum_{k=1}^{m} \boldsymbol{y}_k \boldsymbol{a}_k \boldsymbol{a}_k^\top\right] = \underbrace{\|\boldsymbol{x}^\star\|_2^2 \, \boldsymbol{I} + 2\boldsymbol{x}^\star \boldsymbol{x}^{\star\top}}_{\text{leading eigenvector: } \pm\boldsymbol{x}^\star}$$

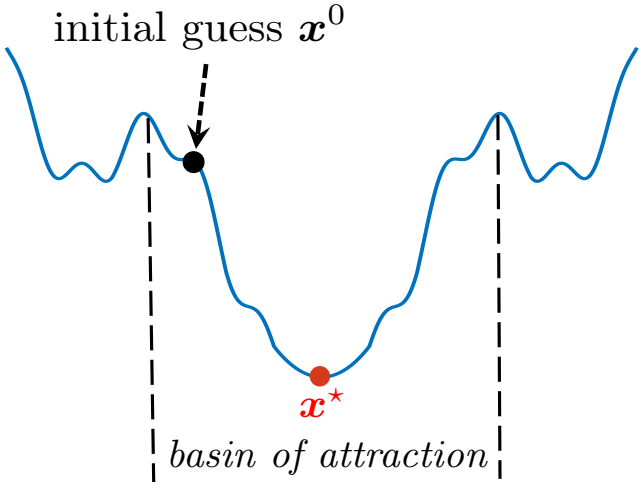
Rationale of two-stage approach

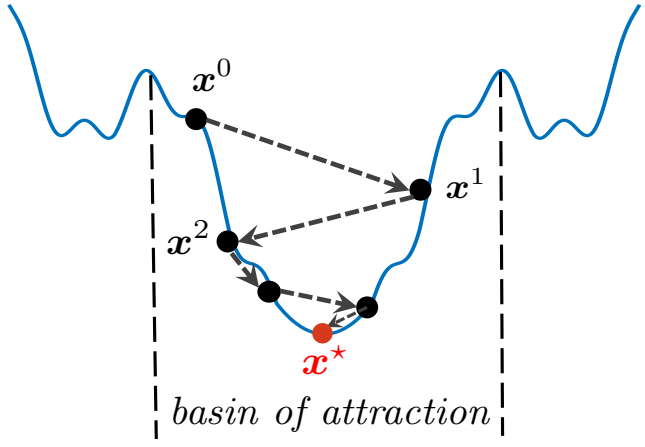


1. initialize within $\underbrace{\text{local basin sufficiently close to } \boldsymbol{x}^\star}$

(restricted) strongly convex; no saddles / spurious local mins

Rationale of two-stage approach





1. initialize within $\underbrace{\text{local basin sufficiently close to } \boldsymbol{x}^\star}$

   (restricted) strongly convex; no saddles / spurious local mins

2. iterative refinement

A highly incomplete list of two-stage methods

phase retrieval:

- Netrapalli, Jain, Sanghavi '13
- Candès, Li, Soltanolkotabi '14
- Chen, Candès '15
- Cai, Li, Ma '15
- Wang, Giannakis, Eldar '16
- Zhang, Zhou, Liang, Chi '16
- Kolte, Ozgur '16
- Zhang, Chi, Liang '16
- Soltanolkotabi '17
- Vaswani, Nayer, Eldar '16
- Chi, Lu '16
- Wang, Zhang, Giannakis, Akcakaya, Chen '16
- Tan, Vershynin '17
- Ma, Wang, Chi, Chen '17
- Duchi, Ruan '17
- Jeong, Gunturk '17
- Yang, Yang, Fang, Zhao, Wang, Neykov '17
- Qu, Zhang, Wright '17
- Goldstein, Studer '16
- Bahmani, Romberg '16
- Hand, Voroninski '16
- Wang, Giannakis, Saad, Chen '17
- Barmherzig, Sun '17
- ...

other problems:

- Keshavan, Montanari, Oh '09
- Sun, Luo '14
- Chen, Wainwright '15
- Tu, Boczar, Simchowitz, Soltanolkotabi, Recht '15
- Zheng, Lafferty '15
- Balakrishnan, Wainwright, Yu '14
- Chen, Suh '15
- Chen, Candès '16
- Li, Ling, Strohmer, Wei '16
- Yi, Park, Chen, Caramanis '16
- Jin, Kakade, Netrapalli '16
- Huang, Kakade, Kong, Valiant '16
- Ling, Strohmer '17
- Li, Ma, Chen, Chi '18
- Aghasi, Ahmed, Hand '17
- Lee, Tian, Romberg '17
- Li, Chi, Zhang, Liang '17
- Cai, Wang, Wei '17
- Abbe, Bandeira, Hall '14
- Chen, Kamath, Suh, Tse '16
- Zhang, Zhou '17
- Boumal '16
- Zhong, Boumal '17
- ...

First theory of WF

$$\mathsf{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) := \min\{\|\boldsymbol{x}^t \pm \boldsymbol{x}^\star\|_2\}$$

Theorem 1 (Candès, Li, Soltanolkotabi '14)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

$$\mathsf{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \lesssim \left(1 - \frac{\eta}{4}\right)^{t/2} \|\boldsymbol{x}^\star\|_2,$$

with high prob., provided that step size $\eta \lesssim 1/n$ *and sample size:* $m \gtrsim n \log n$.

First theory of WF

$$\mathsf{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) := \min\{\|\boldsymbol{x}^t \pm \boldsymbol{x}^\star\|_2\}$$

Theorem 1 (Candès, Li, Soltanolkotabi '14)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

$$\mathsf{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \lesssim \left(1 - \frac{\eta}{4}\right)^{t/2} \|\boldsymbol{x}^\star\|_2,$$

with high prob., provided that step size $\eta \lesssim 1/n$ *and sample size:* $m \gtrsim n \log n$.

- Iteration complexity: $O\big(n \log \frac{1}{\epsilon}\big)$

First theory of WF

$$\mathsf{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) := \min\{\|\boldsymbol{x}^t \pm \boldsymbol{x}^\star\|_2\}$$

Theorem 1 (Candès, Li, Soltanolkotabi '14)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

$$\mathsf{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \lesssim \left(1 - \frac{\eta}{4}\right)^{t/2} \|\boldsymbol{x}^\star\|_2,$$

with high prob., provided that step size $\eta \lesssim 1/n$ *and sample size:* $m \gtrsim n \log n$.

- Iteration complexity: $O(n \log \frac{1}{\epsilon})$
- Sample complexity: $O(n \log n)$

First theory of WF

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) := \min\{\|\boldsymbol{x}^t \pm \boldsymbol{x}^\star\|_2\}$$

Theorem 1 (Candès, Li, Soltanolkotabi '14)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \lesssim \left(1 - \frac{\eta}{4}\right)^{t/2} \|\boldsymbol{x}^\star\|_2,$$

with high prob., provided that step size and sample size: .

- Iteration complexity: $O(n \log \frac{1}{\epsilon})$
- Sample complexity: $O(n \log n)$
- Derived based on (worst-case) local geometry

Improved theory of WF

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) := \min\{\|\boldsymbol{x}^t \pm \boldsymbol{x}^\star\|_2\}$$

Theorem 2 (Ma, Wang, Chi, Chen '17)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \lesssim \left(1 - \frac{\eta}{2}\right)^t \|\boldsymbol{x}^\star\|_2$$

with high prob., provided that step size $\eta \asymp 1/\log n$ *and sample size* $m \gtrsim n \log n$.

- Iteration complexity: $O(n \log \frac{1}{\epsilon}) \searrow O(\log n \log \frac{1}{\epsilon})$
- Sample complexity: $O(n \log n)$
- Derived based on finer analysis of GD trajectory

What does optimization theory say about WF?

Gaussian designs: $\boldsymbol{a}_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n), \quad 1 \le k \le m$

What does optimization theory say about WF?

Gaussian designs: $\boldsymbol{a}_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n), \quad 1 \le k \le m$

Finite-sample level ($m \asymp n \log n$)

$$\nabla^2 f(\boldsymbol{x}) \succ \boldsymbol{0}$$

What does optimization theory say about WF?

$$\textit{Gaussian designs: } \boldsymbol{a}_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n), \quad 1 \le k \le m$$

Finite-sample level ($m \asymp n \log n$)

$$\nabla^2 f(\boldsymbol{x}) \succ \boldsymbol{0} \quad \underbrace{\text{but ill-conditioned}}_{\text{condition number } \asymp n} \text{ (even locally)}$$

What does optimization theory say about WF?

$$\textit{Gaussian designs: } \boldsymbol{a}_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n), \quad 1 \le k \le m$$

Finite-sample level ($m \asymp n \log n$)

$$\nabla^2 f(\boldsymbol{x}) \succ \boldsymbol{0} \quad \underbrace{\text{but ill-conditioned}}_{\text{condition number } \asymp n} \text{ (even locally)}$$

Consequence (Candès et al '14): WF attains ε-accuracy within $O\big(n \log \frac{1}{\varepsilon}\big)$ iterations if $m \asymp n \log n$

Generic optimization theory gives pessimistic bounds

WF converges in $O(n)$ iterations

Generic optimization theory gives pessimistic bounds

WF converges in $O(n)$ iterations

Step size taken to be $\eta = O(1/n)$

Generic optimization theory gives pessimistic bounds

WF converges in $O(n)$ iterations

Step size taken to be $\eta = O(1/n)$

This choice is suggested by worst-case optimization theory

Generic optimization theory gives pessimistic bounds

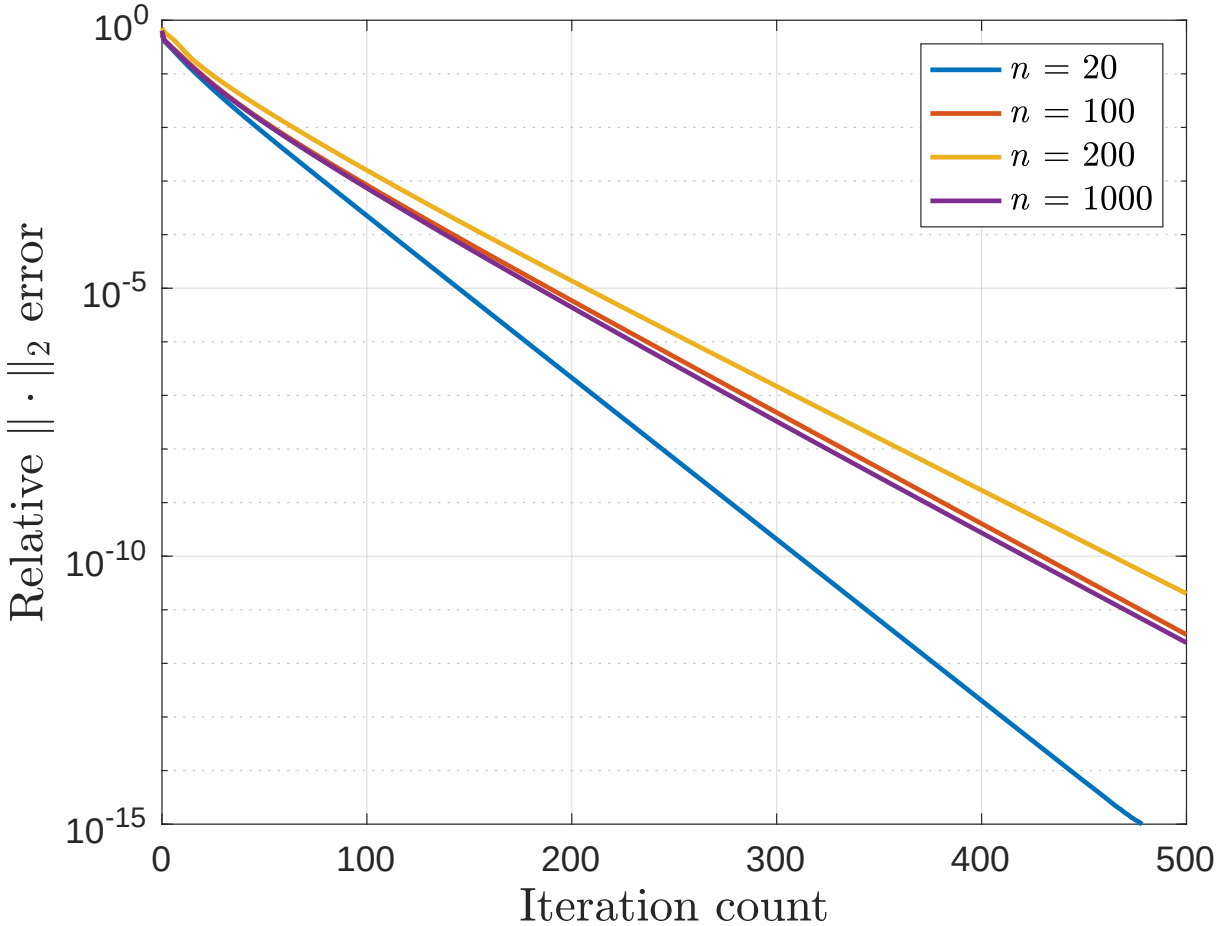
WF converges in $O(n)$ iterations

Step size taken to be $\eta = O(1/n)$

This choice is suggested by worst-case optimization theory

Does it capture what really happens?

Numerical efficiency with $\eta_t = 0.1$



Vanilla GD (WF) converges fast for a constant step size!

A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

$$\nabla^2 f(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^{m}\left[3\left(\boldsymbol{a}_k^\top \boldsymbol{x}\right)^2 - \left(\boldsymbol{a}_k^\top \boldsymbol{x}^\star\right)^2\right]\boldsymbol{a}_k\boldsymbol{a}_k^\top$$

A second look at gradient descent theory

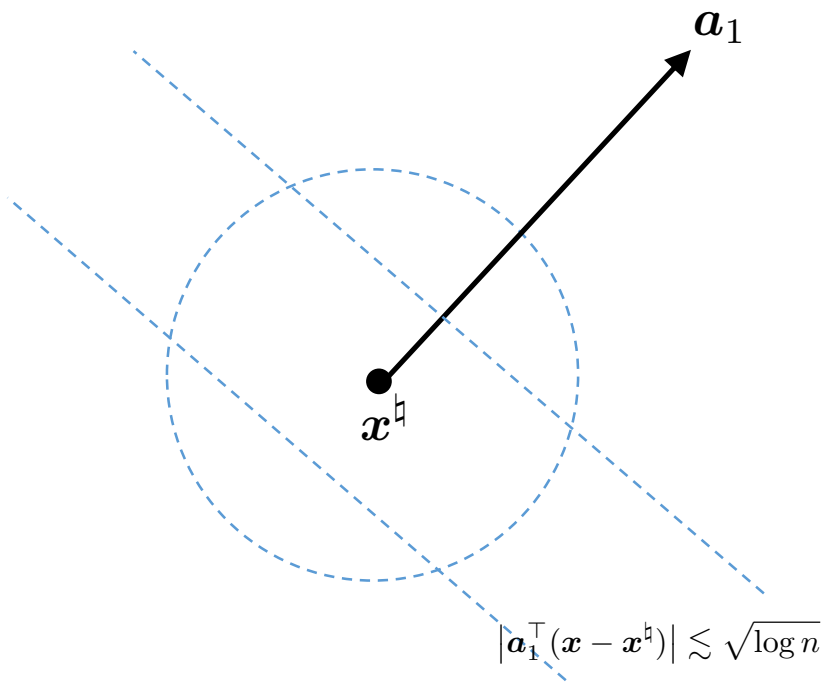
Which local region enjoys both strong convexity and smoothness?

$$\nabla^2 f(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^{m}\left[3\left(\boldsymbol{a}_k^\top \boldsymbol{x}\right)^2 - \left(\boldsymbol{a}_k^\top \boldsymbol{x}^\star\right)^2\right]\boldsymbol{a}_k\boldsymbol{a}_k^\top$$

- Not sufficiently smooth if $\boldsymbol{x}$ and $\boldsymbol{a}_k$ are too close (coherent)

A second look at gradient descent theory

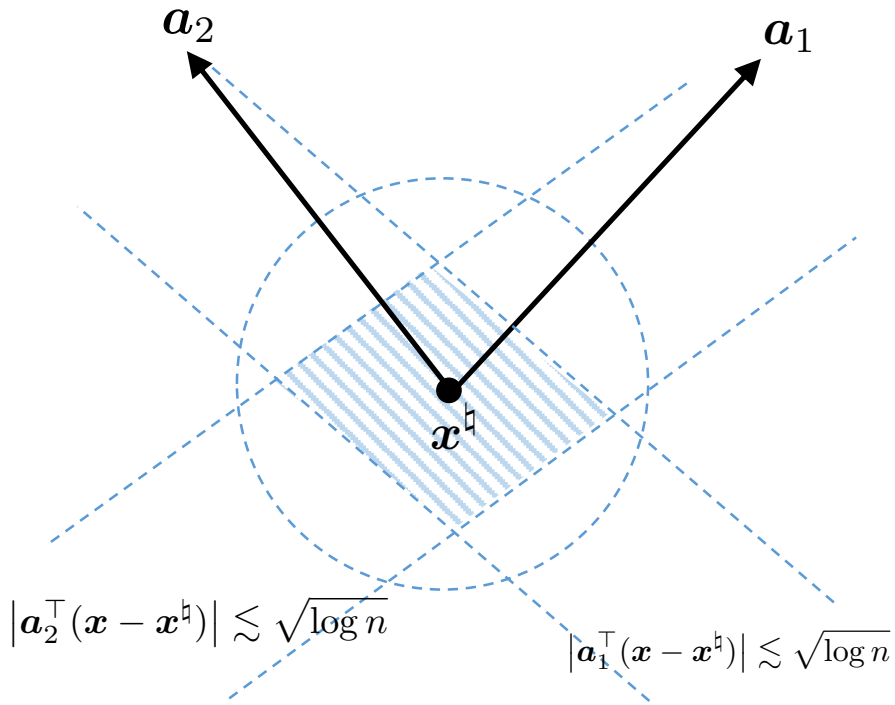
Which local region enjoys both strong convexity and smoothness?



- $\boldsymbol{x}$ is incoherent w.r.t. sampling vectors $\{\boldsymbol{a}_k\}$ (incoherence region)

A second look at gradient descent theory

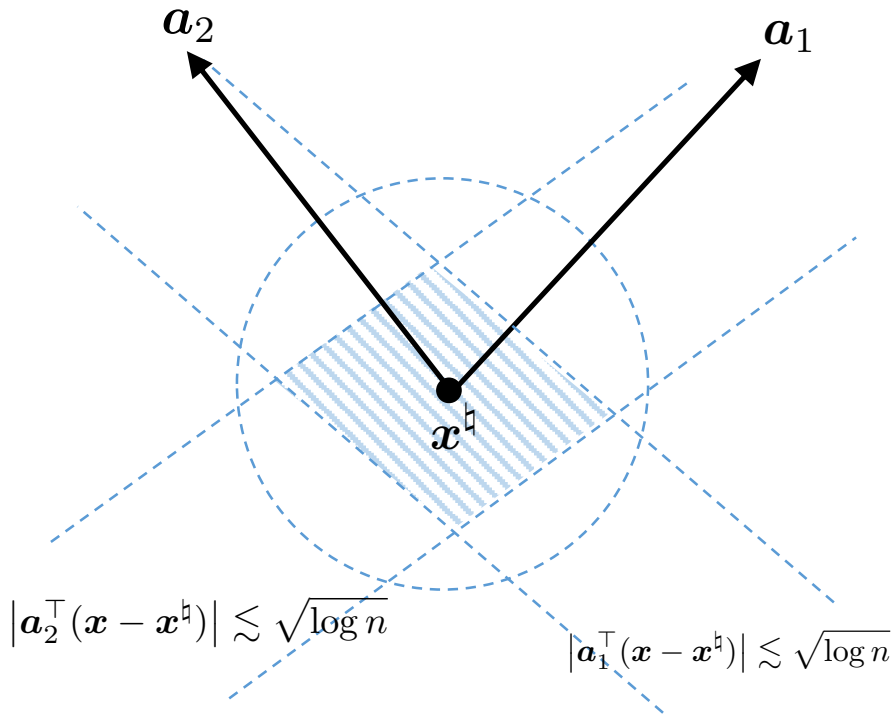
Which local region enjoys both strong convexity and smoothness?



- $\boldsymbol{x}$ is incoherent w.r.t. sampling vectors $\{\boldsymbol{a}_k\}$ (incoherence region)

A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

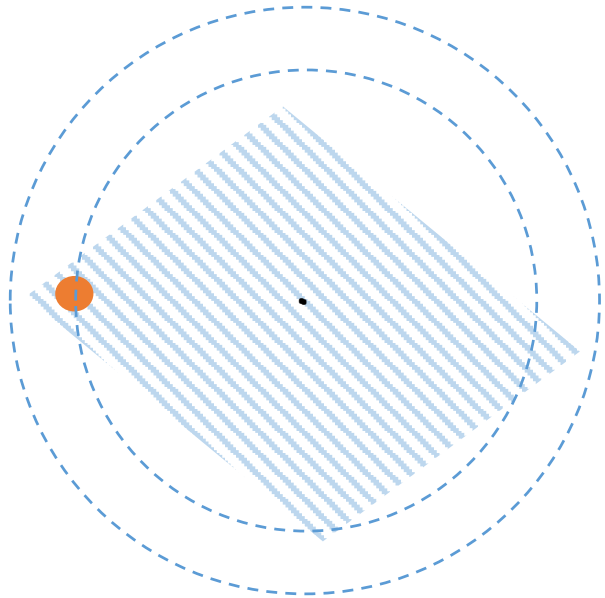


- $\boldsymbol{x}$ is incoherent w.r.t. sampling vectors $\{\boldsymbol{a}_k\}$ (incoherence region)

Prior works suggest enforcing regularization (e.g. truncation, projection, regularized loss) to promote incoherence

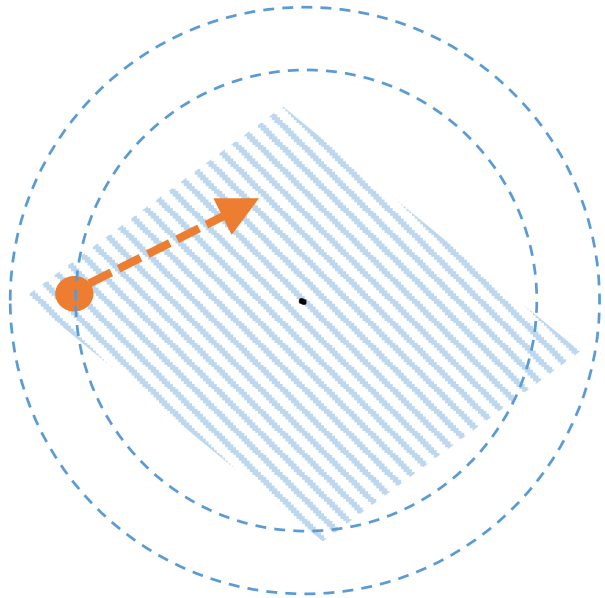
Encouraging message: GD is implicitly regularized

region of local strong convexity + smoothness



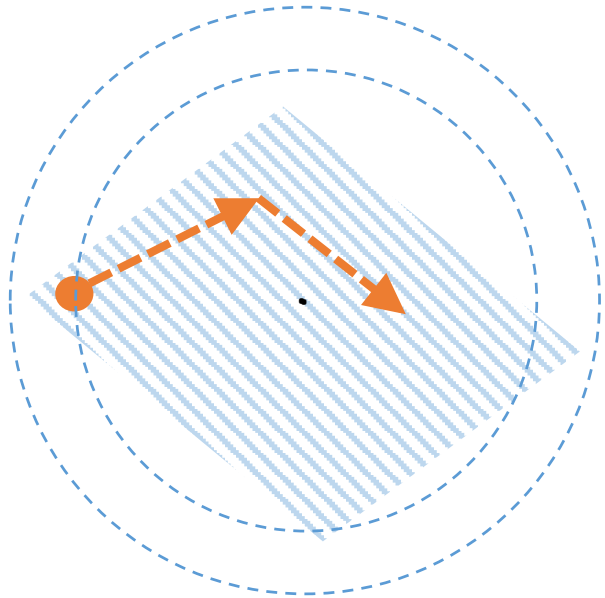
Encouraging message: GD is implicitly regularized

region of local strong convexity + smoothness



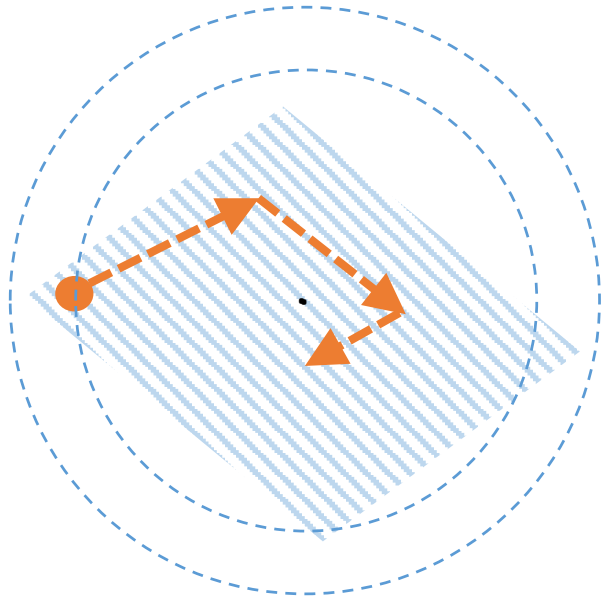
Encouraging message: GD is implicitly regularized

region of local strong convexity + smoothness



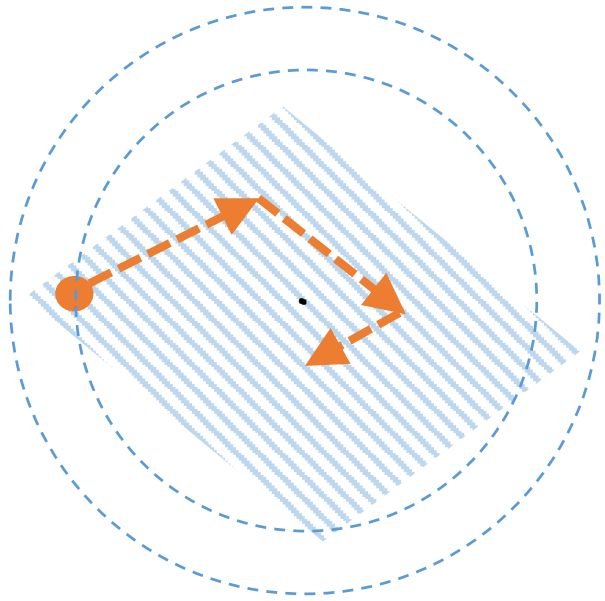
Encouraging message: GD is implicitly regularized

region of local strong convexity + smoothness



Encouraging message: GD is implicitly regularized

region of local strong convexity + smoothness



GD implicitly forces iterates to remain incoherent with $\{\boldsymbol{a}_k\}$

$$\max_k \left|\boldsymbol{a}_k^\top(\boldsymbol{x}^t - \boldsymbol{x}^\star)\right| \lesssim \sqrt{\log n}\,\|\boldsymbol{x}^\star\|_2, \quad \forall t$$

— cannot be derived from generic optimization theory; relies on finer statistical analysis for entire trajectory of GD

Theoretical guarantees for local refinement stage

Theorem 3 (Ma, Wang, Chi, Chen '17)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

- $\max_k \left|\boldsymbol{a}_k^\top \boldsymbol{x}^t\right| \lesssim \sqrt{\log n}\, \|\boldsymbol{x}^\star\|_2$ *(incoherence)*

Theoretical guarantees for local refinement stage

Theorem 3 (Ma, Wang, Chi, Chen '17)

Under i.i.d. Gaussian design, WF with spectral initialization achieves

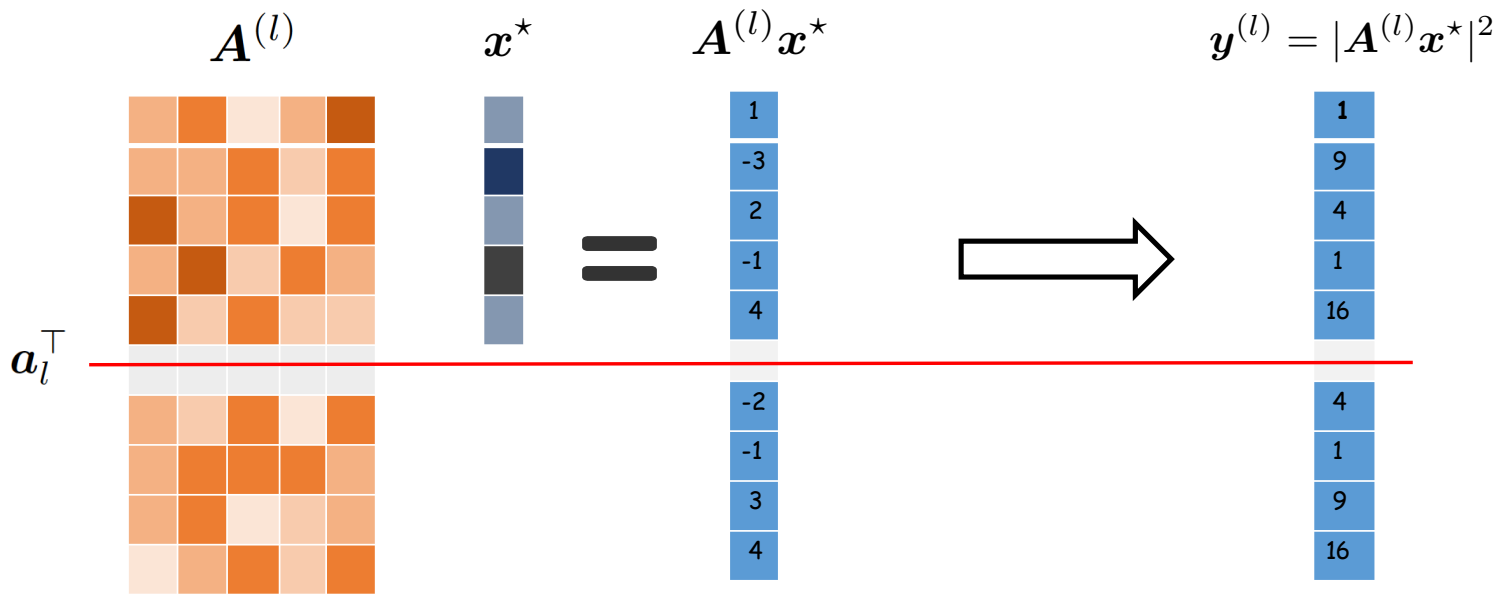
- $\max_k \left|\boldsymbol{a}_k^\top \boldsymbol{x}^t\right| \lesssim \sqrt{\log n}\, \|\boldsymbol{x}^\star\|_2$ *(incoherence)*
- $\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \lesssim \left(1 - \frac{\eta}{2}\right)^t \|\boldsymbol{x}^\star\|_2$ *(linear convergence)*

provided that step size $\eta \asymp 1/\log n$ *and sample size* $m \gtrsim n \log n$.

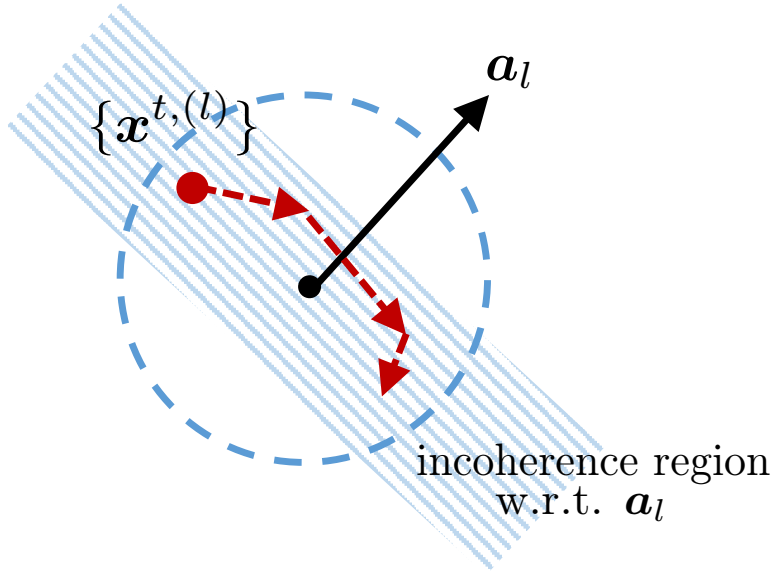
- Attains ε accuracy within $O\big(\log n \, \log \frac{1}{\varepsilon}\big)$ iterations

Key proof idea: leave-one-out analysis

For each $1 \le l \le m$, introduce leave-one-out iterates $\boldsymbol{x}^{t,(l)}$ by dropping lth measurement

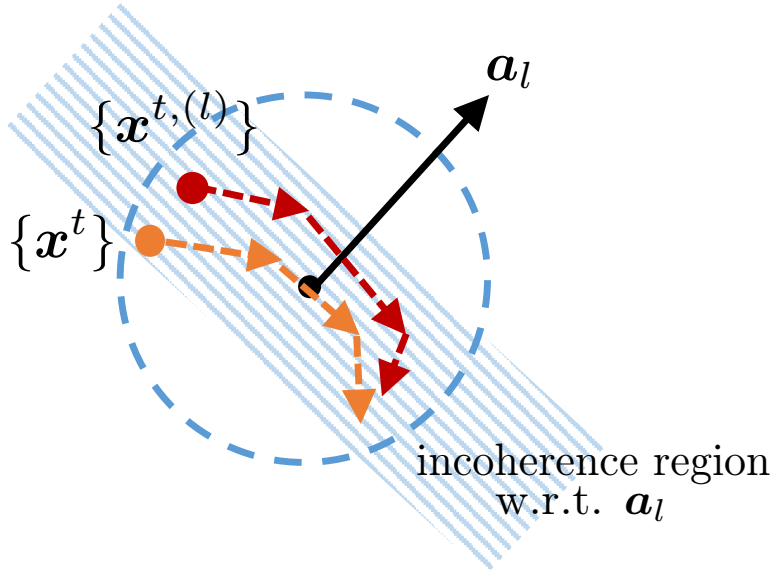


Key proof idea: leave-one-out analysis



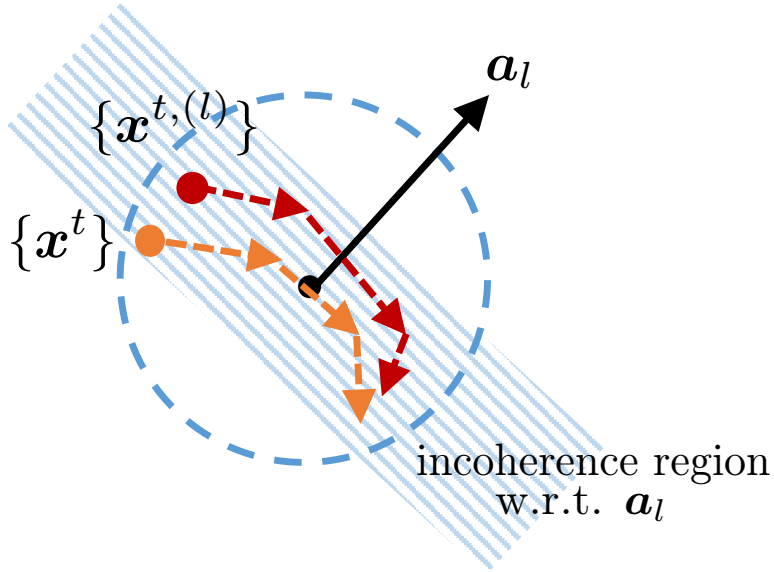
- Leave-one-out iterate $\boldsymbol{x}^{t,(l)}$ is independent of $\boldsymbol{a}_l$

Key proof idea: leave-one-out analysis



- Leave-one-out iterate $\boldsymbol{x}^{t,(l)}$ is independent of $\boldsymbol{a}_l$
- Leave-one-out iterate $\boldsymbol{x}^{t,(l)} \approx$ true iterate $\boldsymbol{x}^t$

Key proof idea: leave-one-out analysis

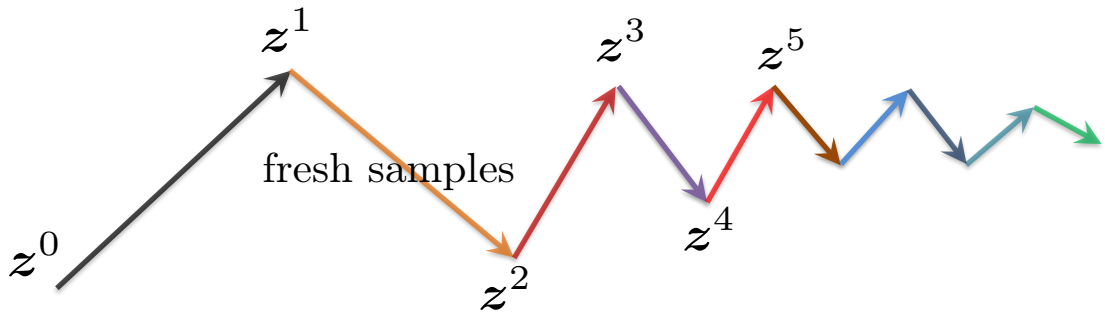


- Leave-one-out iterate $\boldsymbol{x}^{t,(l)}$ is independent of $\boldsymbol{a}_l$
- Leave-one-out iterate $\boldsymbol{x}^{t,(l)} \approx$ true iterate $\boldsymbol{x}^t$

$$\Longrightarrow \quad \boldsymbol{x}^t \text{ is } \underbrace{\text{nearly independent of}}_{\text{nearly orthogonal to}} \boldsymbol{a}_l$$

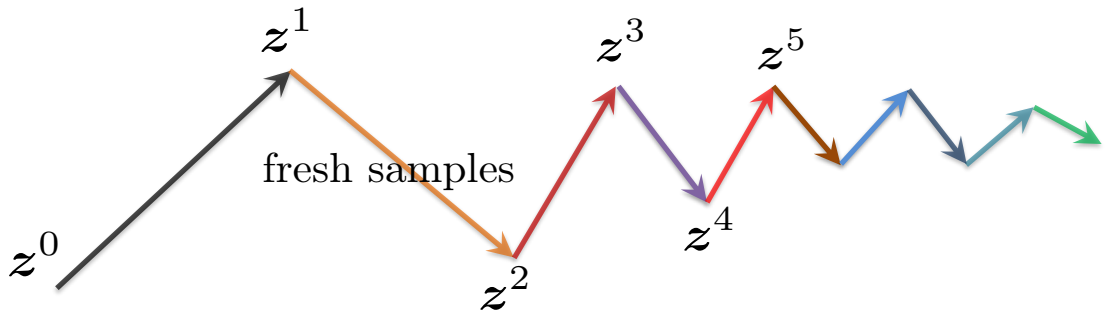
No need of sample splitting

- Several prior works use sample-splitting: require fresh samples at each iteration; not practical but helps analysis

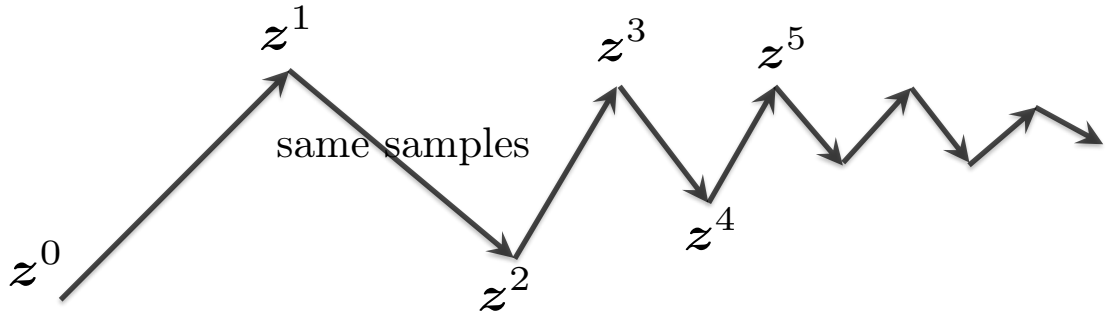


No need of sample splitting

- Several prior works use sample-splitting: require fresh samples at each iteration; not practical but helps analysis



- **This tutorial:** reuses all samples in all iterations



Other examples: low-rank matrix estimation

Low-rank matrix completion

Complete $\boldsymbol{M}$ from partial entries $M_{i,j},\ (i,j) \in \Omega$

where (i,j) is included in Ω independently with prob. p

find low-rank $\widehat{\boldsymbol{M}}$ s.t. $\mathcal{P}_\Omega(\widehat{\boldsymbol{M}}) = \mathcal{P}_\Omega(\boldsymbol{M})$

In matrix completion, strong convexity and smoothness do not hold in general

$\rightarrow$ need to regularize the loss function by promoting **incoherent** solutions

Incoherence for matrix completion

Definition 4 (Incoherence for matrix completion)

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

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

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

Gradient descent for matrix completion

Let $\boldsymbol{M} = \boldsymbol{X}^\star \boldsymbol{X}^{\star\top}$. Observe

$$Y_{i,j} = M_{i,j} + E_{i,j}, \quad (i,j) \in \Omega$$

where $(i,j) \in \Omega$ independently with prob. p, and $E_{i,j} \sim \mathcal{N}(0, \sigma^2)$[1]

$$\text{minimize} \left\| \mathcal{P}_\Omega(\widehat{\boldsymbol{M}} - \boldsymbol{Y}) \right\|_{\mathrm{F}}^2 \quad \text{s.t.} \quad \text{rank}(\widehat{\boldsymbol{M}}) \le r$$

[1] can be relaxed to sub-Gaussian noise and the asymmetric case

Gradient descent for matrix completion

Let $\boldsymbol{M} = \boldsymbol{X}^\star \boldsymbol{X}^{\star\top}$. Observe

$$Y_{i,j} = M_{i,j} + E_{i,j}, \quad (i,j) \in \Omega$$

where $(i,j) \in \Omega$ independently with prob. p, and $E_{i,j} \sim \mathcal{N}(0, \sigma^2)$[1]

$$\text{minimize} \left\| \mathcal{P}_\Omega(\widehat{\boldsymbol{M}} - \boldsymbol{Y}) \right\|_{\mathrm{F}}^2 \quad \text{s.t.} \quad \text{rank}(\widehat{\boldsymbol{M}}) \le r$$

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad \underbrace{f(\boldsymbol{X}) = \sum_{(j,k) \in \Omega} \left(\boldsymbol{e}_j^\top \boldsymbol{X} \boldsymbol{X}^\top \boldsymbol{e}_k - Y_{j,k} \right)^2}_{\text{unregularized least-squares loss}}$$

[1] can be relaxed to sub-Gaussian noise and the asymmetric case

Gradient descent for matrix completion

(1) **Spectral initialization**: let $\boldsymbol{U}^0\boldsymbol{\Sigma}^0\boldsymbol{U}^{0\top}$ be rank-r eigendecomposition of

$$\frac{1}{p}\mathcal{P}_{\Omega}(\boldsymbol{Y}).$$

and set $\boldsymbol{X}^0 = \boldsymbol{U}^0\left(\boldsymbol{\Sigma}^0\right)^{1/2}$

(2) **Gradient descent updates:**

$$\boldsymbol{X}^{t+1} = \boldsymbol{X}^t - \eta_t \nabla f\left(\boldsymbol{X}^t\right), \qquad t = 0, 1, \cdots$$

Gradient descent for matrix completion

Define the optimal transform from the tth iterate $\boldsymbol{X}^t$ to $\boldsymbol{X}^\star$ as

$$\boldsymbol{Q}^t := \mathsf{argmin}_{\boldsymbol{R} \in \mathcal{O}^{r \times r}} \left\| \boldsymbol{X}^t \boldsymbol{R} - \boldsymbol{X}^\star \right\|_{\mathrm{F}}$$

where $\mathcal{O}^{r \times r}$ is the set of $r \times r$ orthonormal matrices

- orthogonal Procrustes problem

Gradient descent for matrix completion

Theorem 5 (Noiseless MC, Ma, Wang, Chi, Chen '17)

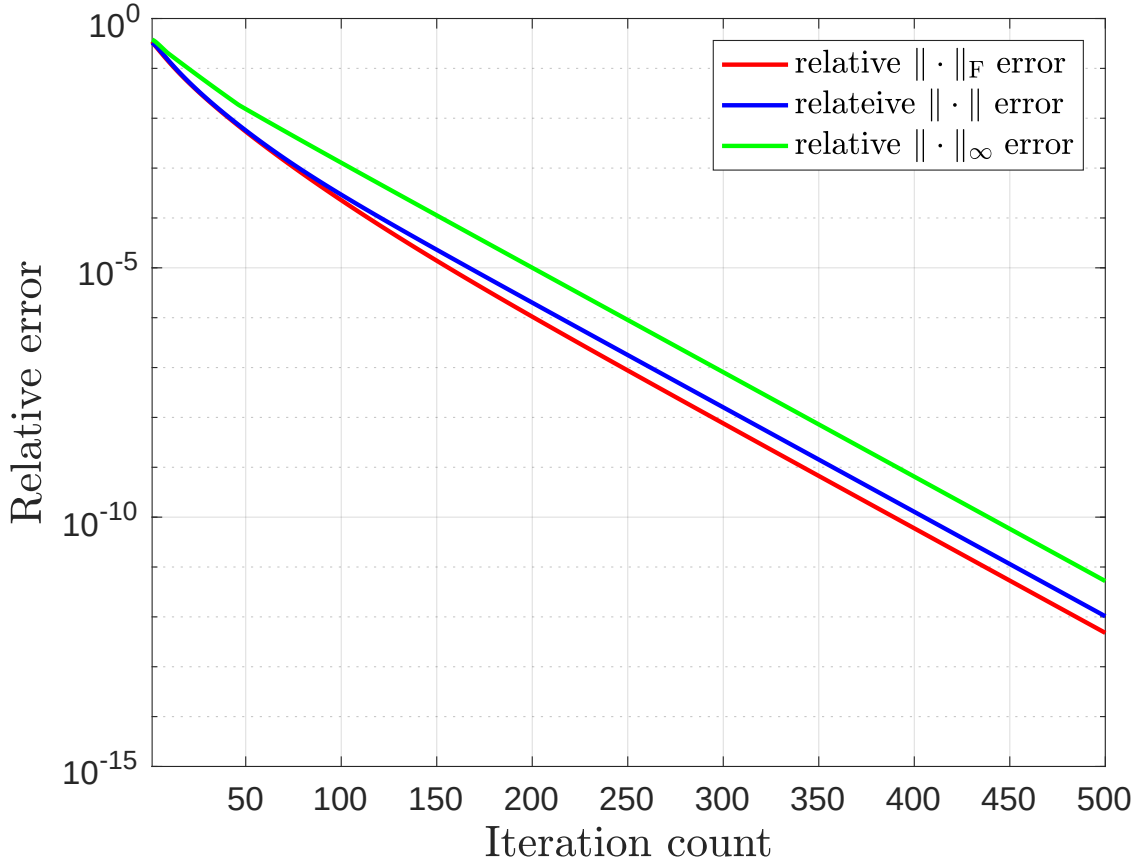
Suppose $\boldsymbol{M} = \boldsymbol{X}^\star \boldsymbol{X}^{\star\top}$ *is rank-*r*, incoherent and well-conditioned. Vanilla GD (with spectral initialization) achieves*

- $\|\boldsymbol{X}^t\boldsymbol{Q}^t - \boldsymbol{X}^\star\|_{\mathrm{F}} \lesssim \rho^t \mu r \frac{1}{\sqrt{np}} \|\boldsymbol{X}^\star\|_{\mathrm{F}}$,
- $\|\boldsymbol{X}^t\boldsymbol{Q}^t - \boldsymbol{X}^\star\| \lesssim \rho^t \mu r \frac{1}{\sqrt{np}} \|\boldsymbol{X}^\star\|$, *(spectral)*
- $\|\boldsymbol{X}^t\boldsymbol{Q}^t - \boldsymbol{X}^\star\|_{2,\infty} \lesssim \rho^t \mu r \sqrt{\frac{\log n}{np}} \|\boldsymbol{X}^\star\|_{2,\infty}$, *(incoherence)*

where $0 < \rho < 1$, *if the step size* $\eta \asymp 1/\sigma_{max}$ *and the sample complexity* $n^2 p \gtrsim \mu^3 n r^3 \log^3 n$

- vanilla gradient descent converges linearly for matrix completion!

Numerical evidence for noiseless data



Relative error of $\boldsymbol{X}^t\boldsymbol{X}^{t\top}$ (measured by $\|\cdot\|_{\mathrm{F}}, \|\cdot\|, \|\cdot\|_{\infty}$) vs. iteration count for MC, where $n = 1000$, $r = 10$, $p = 0.1$, and $\eta_t = 0.2$

Related theory

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \sum_{(j,k) \in \Omega} \left(\boldsymbol{e}_j^\top \boldsymbol{X} \boldsymbol{X}^\top \boldsymbol{e}_k - Y_{j,k} \right)^2$$

Related theory promotes incoherence explicitly:

Related theory

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \sum_{(j,k) \in \Omega} \left(\boldsymbol{e}_j^\top \boldsymbol{X} \boldsymbol{X}^\top \boldsymbol{e}_k - Y_{j,k} \right)^2$$

Related theory promotes incoherence explicitly:

- regularized loss (solve $\min_{\boldsymbol{X}} \; f(\boldsymbol{X}) + Q(\boldsymbol{X})$ instead)
  - e.g. Keshavan, Montanari, Oh '10, Sun, Luo '14, Ge, Lee, Ma '16

Related theory

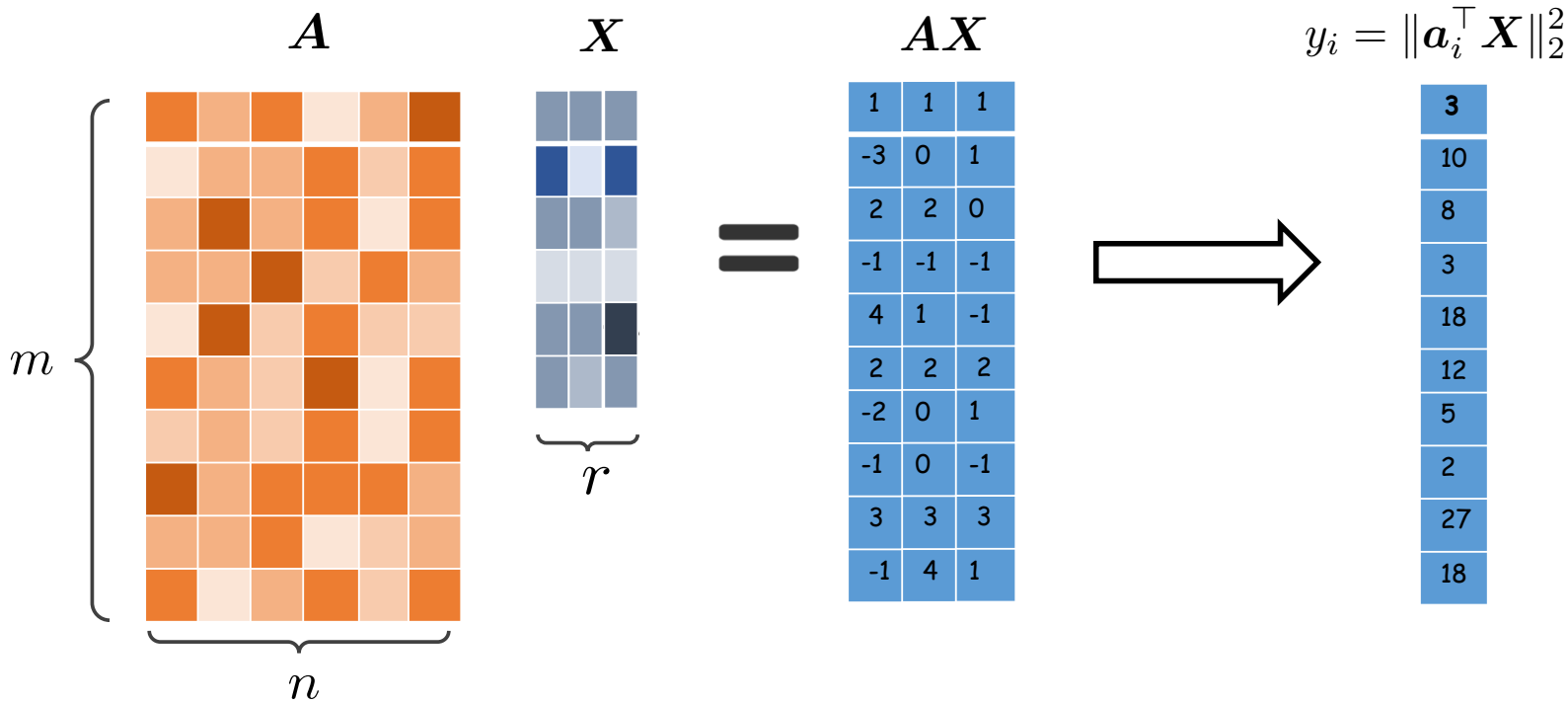
$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \sum_{(j,k) \in \Omega} \left(\boldsymbol{e}_j^\top \boldsymbol{X} \boldsymbol{X}^\top \boldsymbol{e}_k - Y_{j,k} \right)^2$$

Related theory promotes incoherence explicitly:

- regularized loss (solve $\min_{\boldsymbol{X}} \; f(\boldsymbol{X}) + Q(\boldsymbol{X})$ instead)
  - e.g. Keshavan, Montanari, Oh '10, Sun, Luo '14, Ge, Lee, Ma '16
- projection onto set of incoherent matrices
  - e.g. Chen, Wainwright '15, Zheng, Lafferty '16

$$\boldsymbol{X}^{t+1} = \mathcal{P}_{\mathcal{C}} \left(\boldsymbol{X}^t - \eta_t \nabla f\left(\boldsymbol{X}^t\right) \right), \qquad t = 0, 1, \cdots$$

Quadratic sampling



Recover $\boldsymbol{X}^\star \in \mathbb{R}^{n \times r}$ from m random quadratic measurements

$$y_i \quad = \quad \|\boldsymbol{a}_i^\top \boldsymbol{X}^\star\|_2^2, \qquad i = 1, \ldots, m$$

Applications: quantum state tomography, covariance sketching, ...

Gradient descent with spectral initialization

$$\text{minimize}_{\boldsymbol{X} \in \mathbb{R}^{n \times r}} \quad f(\boldsymbol{X}) = \frac{1}{4m} \sum_{k=1}^{m} \left(\|\boldsymbol{a}_k^\top \boldsymbol{X}\|_2^2 - y_k \right)^2$$

Gradient descent with spectral initialization

$$\text{minimize}_{\boldsymbol{X}\in\mathbb{R}^{n\times r}} \quad f(\boldsymbol{X}) = \frac{1}{4m}\sum_{k=1}^{m}\left(\|\boldsymbol{a}_k^\top \boldsymbol{X}\|_2^2 - y_k\right)^2$$

Theorem 6 (Quadratic sampling)

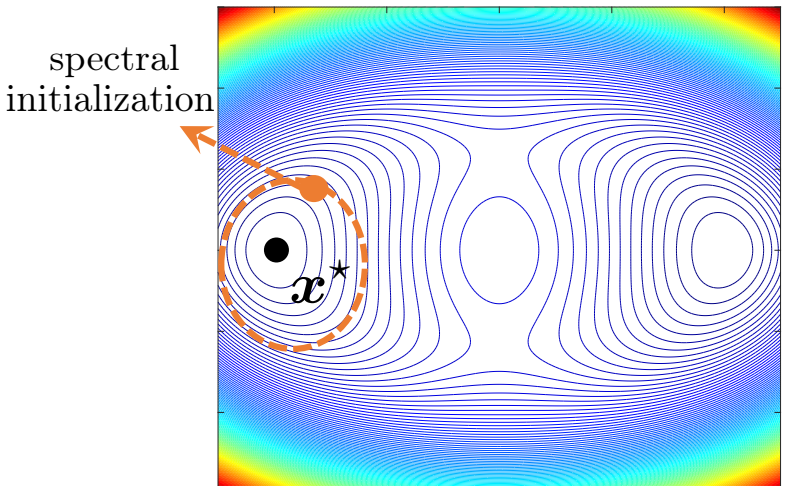
Under i.i.d. Gaussian designs $\boldsymbol{a}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$*, GD (with spectral initialization) achieves*

- $\max_l \left\|\boldsymbol{a}_l^\top(\boldsymbol{X}^t\boldsymbol{Q}^t - \boldsymbol{X}^\star)\right\|_2 \lesssim \sqrt{\log n}\,\frac{\sigma_r^2(\boldsymbol{X}^\star)}{\|\boldsymbol{X}^\star\|_{\mathrm{F}}}$ *(incoherence)*
- $\|\boldsymbol{X}^t\boldsymbol{Q}^t - \boldsymbol{X}^\star\|_{\mathrm{F}} \lesssim \left(1 - \frac{\sigma_r^2(\boldsymbol{X}^\star)\eta}{2}\right)^t \|\boldsymbol{X}^\star\|_{\mathrm{F}}$ *(linear convergence)*

provided that $\eta \asymp \frac{1}{(\log n \vee r)^2\sigma_r^2(\boldsymbol{X}^\star)}$ *and* $m \gtrsim nr^4\log n$

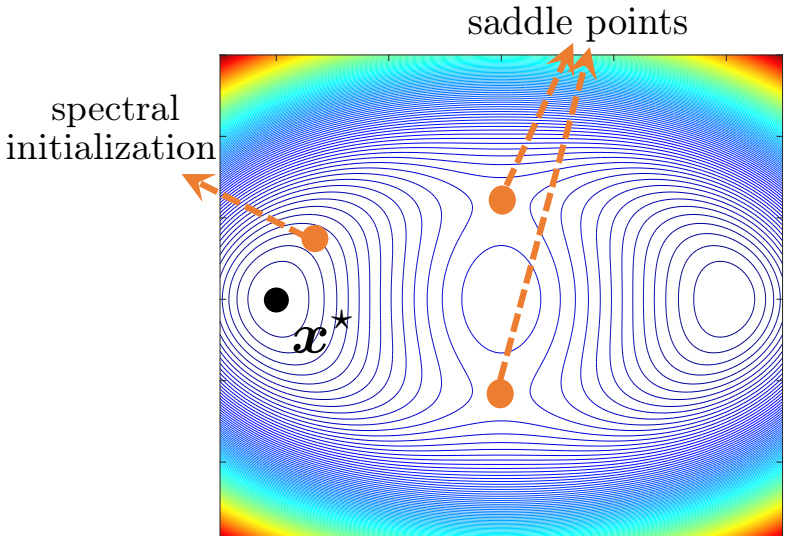
Are carefully-designed initialization or saddle-point escaping schemes necessary for fast convergence?

Initialization



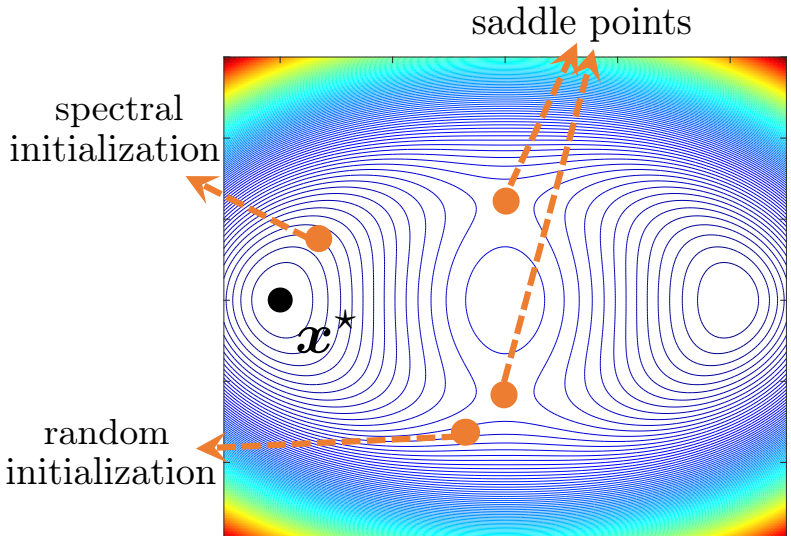
- Spectral initialization gets us reasonably close to truth

Initialization



- Spectral initialization gets us reasonably close to truth
- Cannot initialize GD from anywhere, e.g. it might get stucked at local stationary points (e.g. saddle points)

Initialization

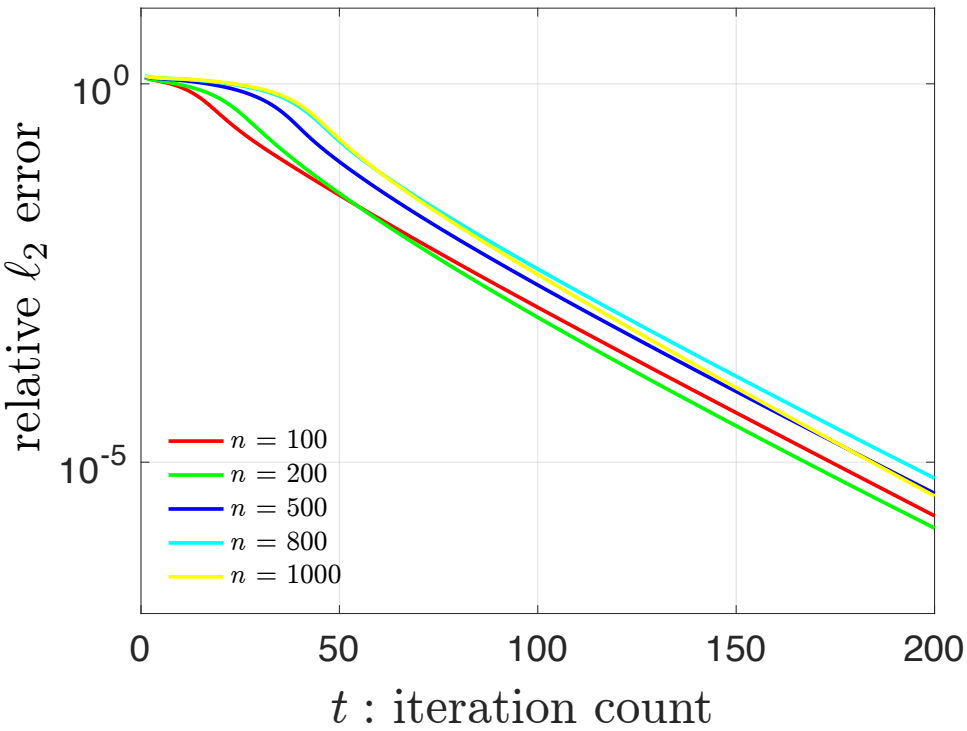


- Spectral initialization gets us reasonably close to truth
- Cannot initialize GD from anywhere, e.g. it might get stucked at local stationary points (e.g. saddle points)

Can we initialize GD randomly, which is simpler and model-agnostic?

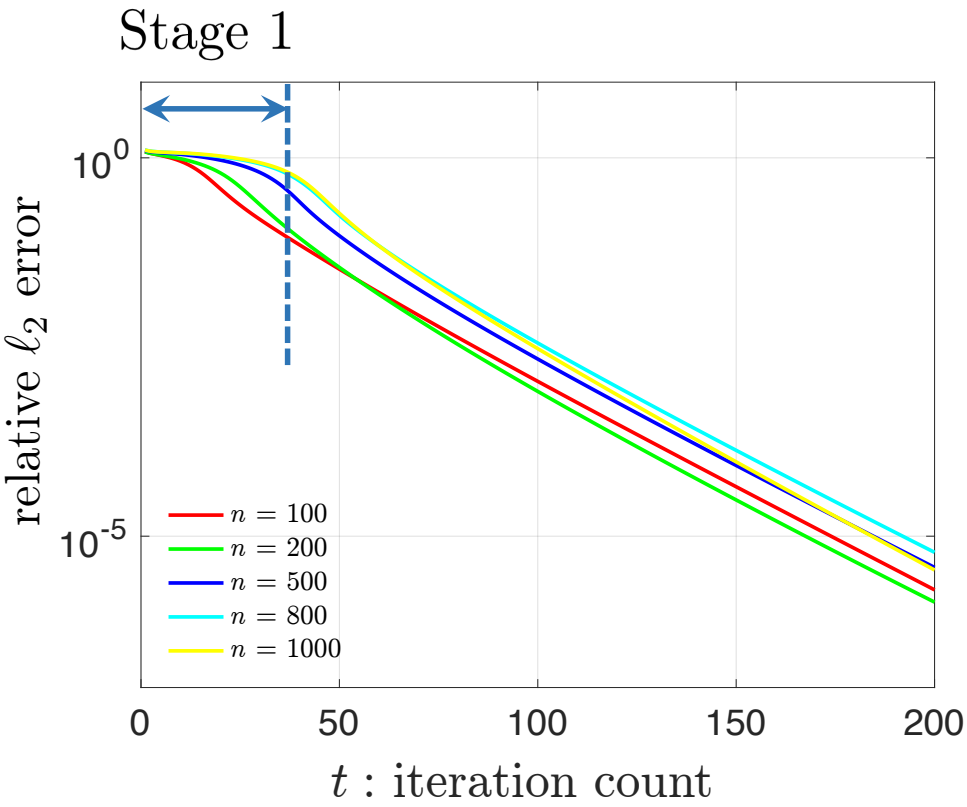
Numerical efficiency of randomly initialized GD

$$\eta_t = 0.1,\ \boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n),\ m = 10n,\ \boldsymbol{x}^0 \sim \mathcal{N}(\boldsymbol{0}, n^{-1}\boldsymbol{I}_n)$$



Numerical efficiency of randomly initialized GD

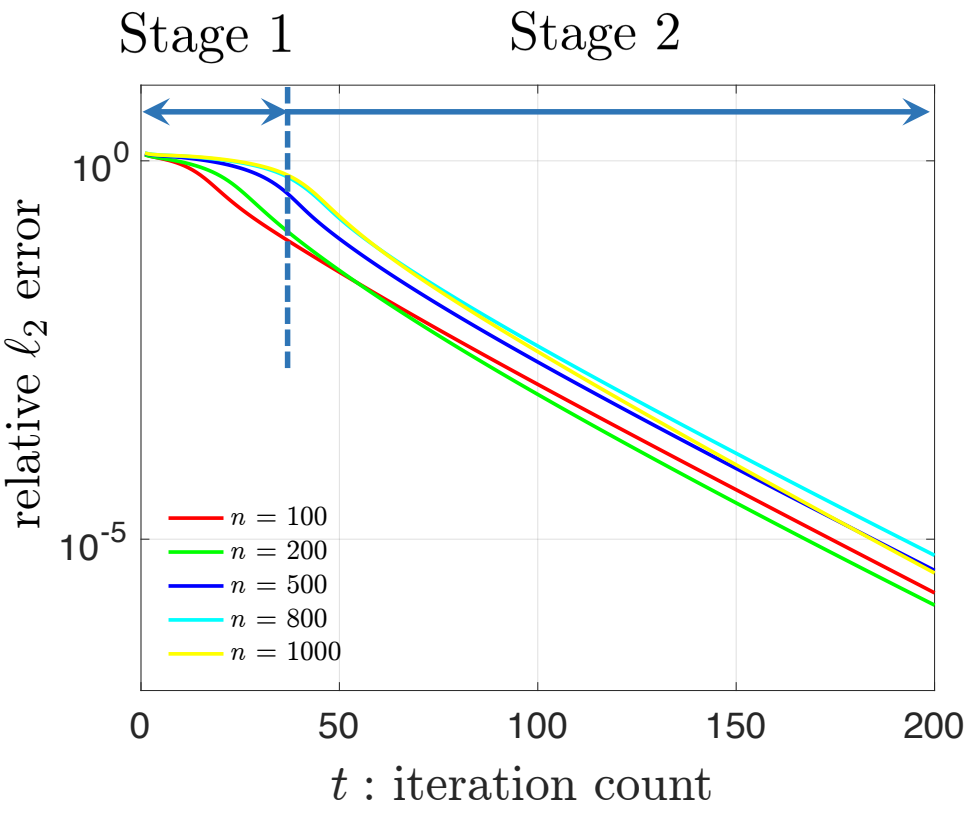
$$\eta_t = 0.1,\ \boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n),\ m = 10n,\ \boldsymbol{x}^0 \sim \mathcal{N}(\boldsymbol{0}, n^{-1}\boldsymbol{I}_n)$$



Randomly initialized GD enters local basin within **a few iterations**

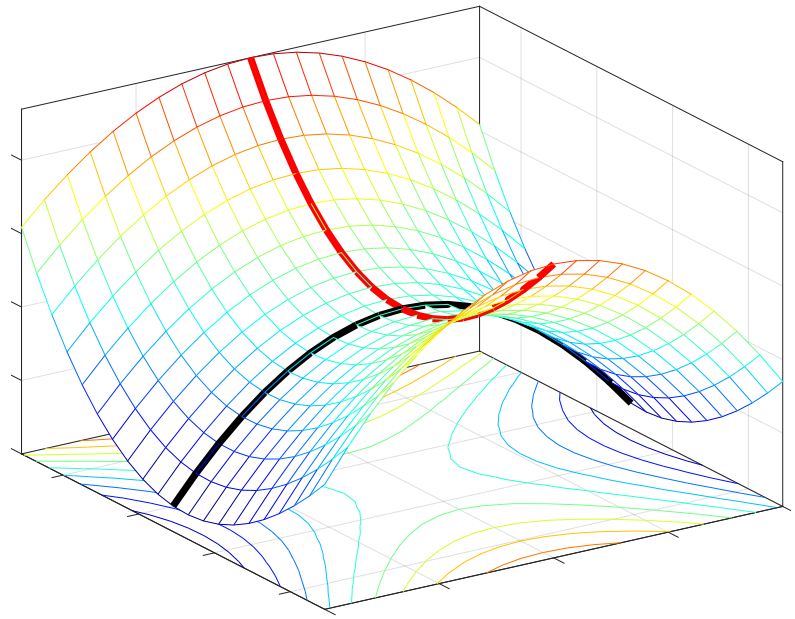
Numerical efficiency of randomly initialized GD

$$\eta_t = 0.1,\ \boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n),\ m = 10n,\ \boldsymbol{x}^0 \sim \mathcal{N}(\boldsymbol{0}, n^{-1}\boldsymbol{I}_n)$$



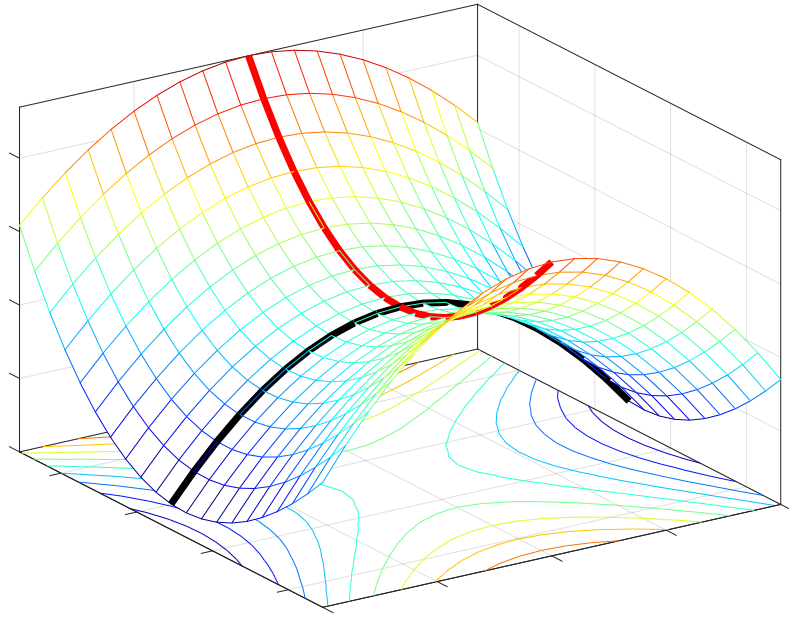
Randomly initialized GD enters local basin within **a few iterations**

A geometric analysis



- if $m \gtrsim n \log^3 n$, then (Sun et al. '16)
  - there is no spurious local mins
  - all saddle points are strict (i.e. associated Hessian matrices have at least one sufficiently negative eigenvalue)

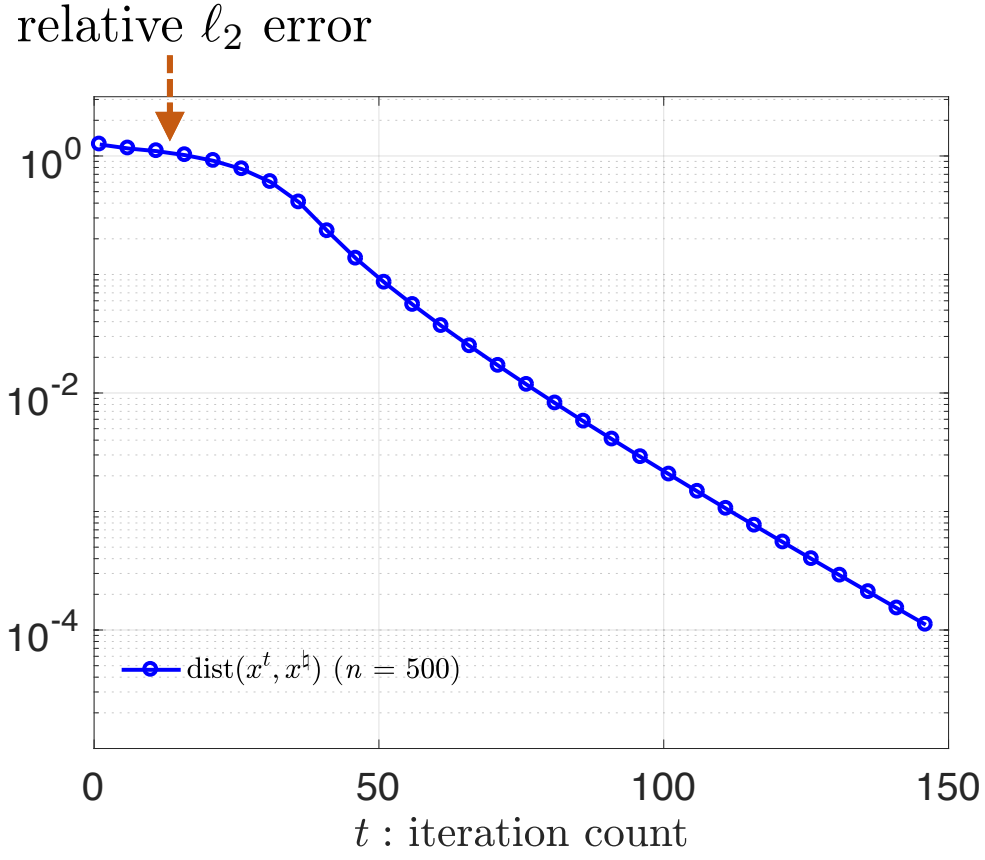
A geometric analysis



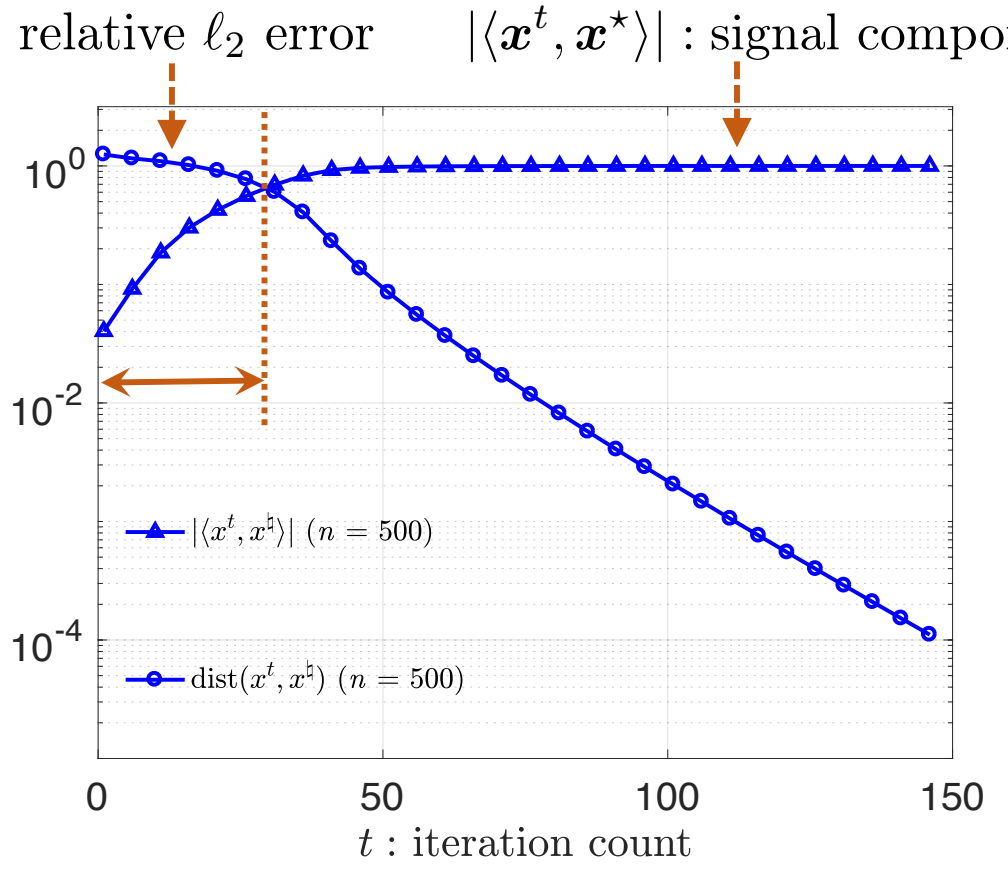
- With such benign landscape, GD with random initialization converges to global min almost surely (Lee et al. '16)

No convergence rate guarantees for vanilla GD!

Exponential growth of signal strength in Stage 1

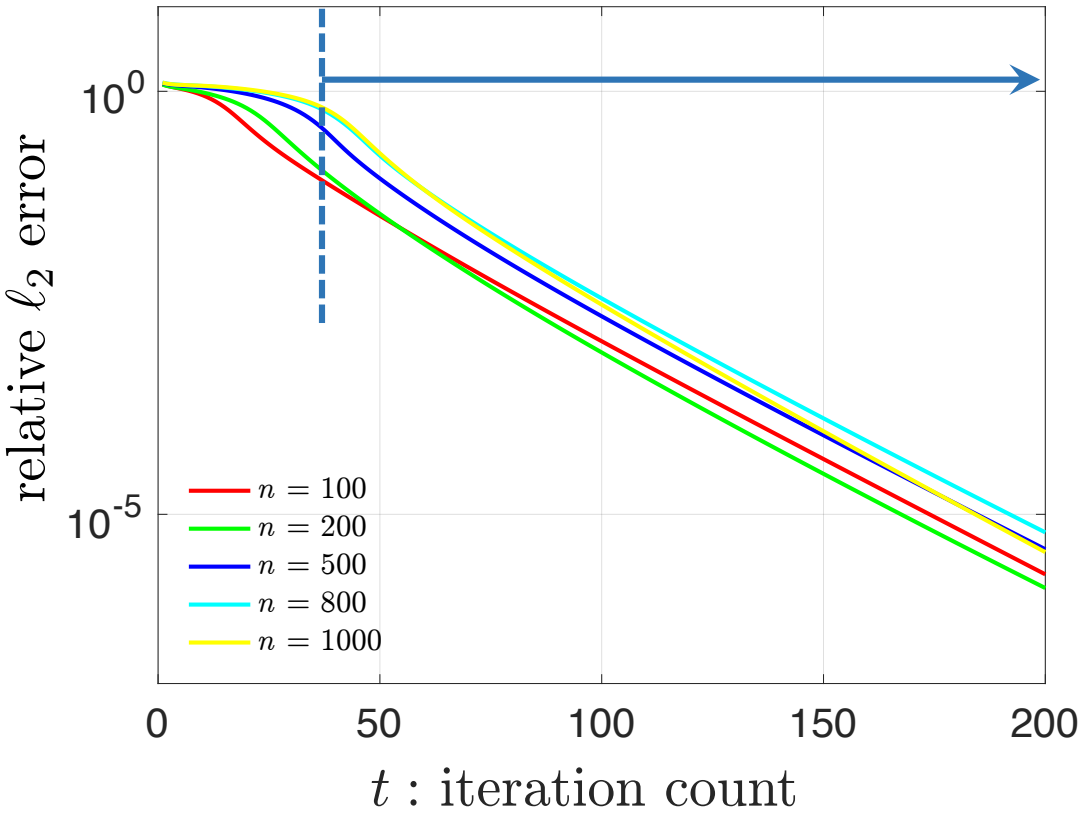


Exponential growth of signal strength in Stage 1

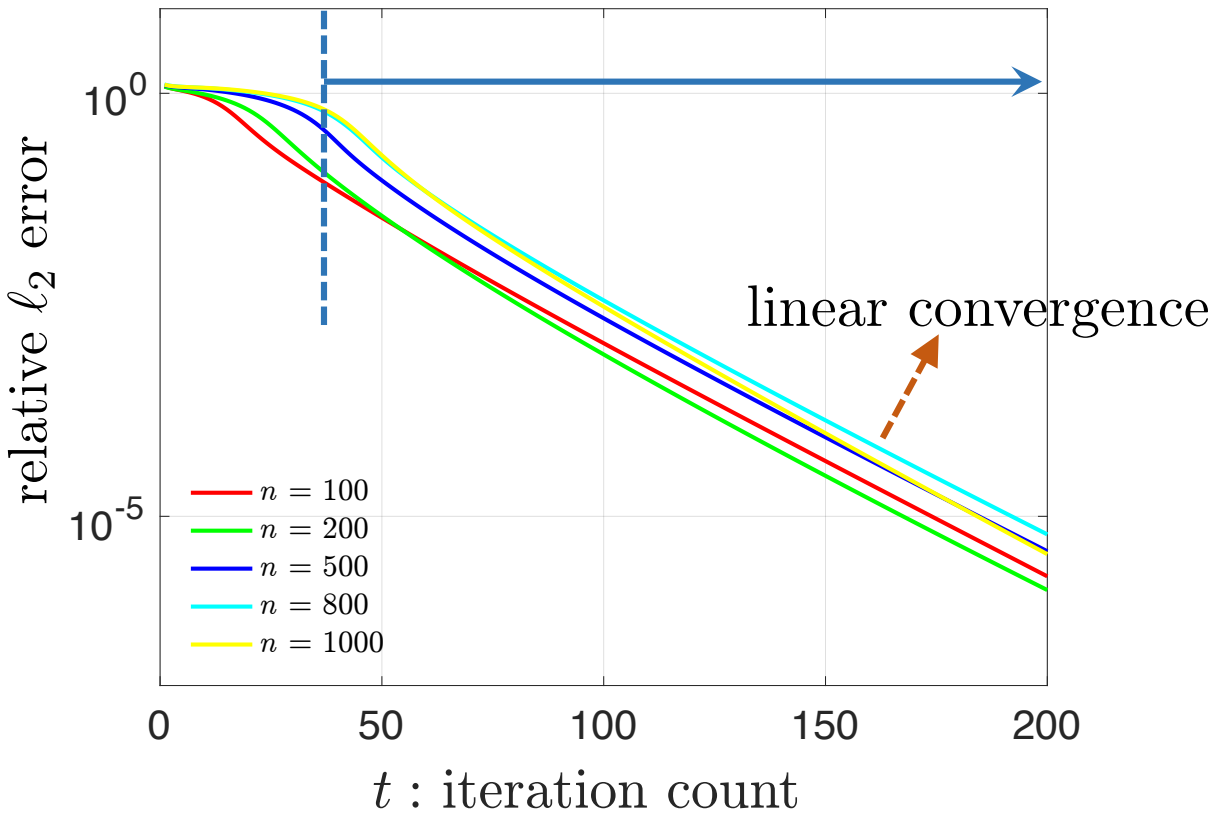


Numerically, $O(\log n)$ iterations are enough to enter local region

Linear / geometric convergence in Stage 2



Linear / geometric convergence in Stage 2



Numerically, GD converges linearly within local region

Theoretical guarantees for randomly initialized GD

These numerical findings can be formalized when $\boldsymbol{a}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$:

Theorem 7 (Chen, Chi, Fan, Ma '18)

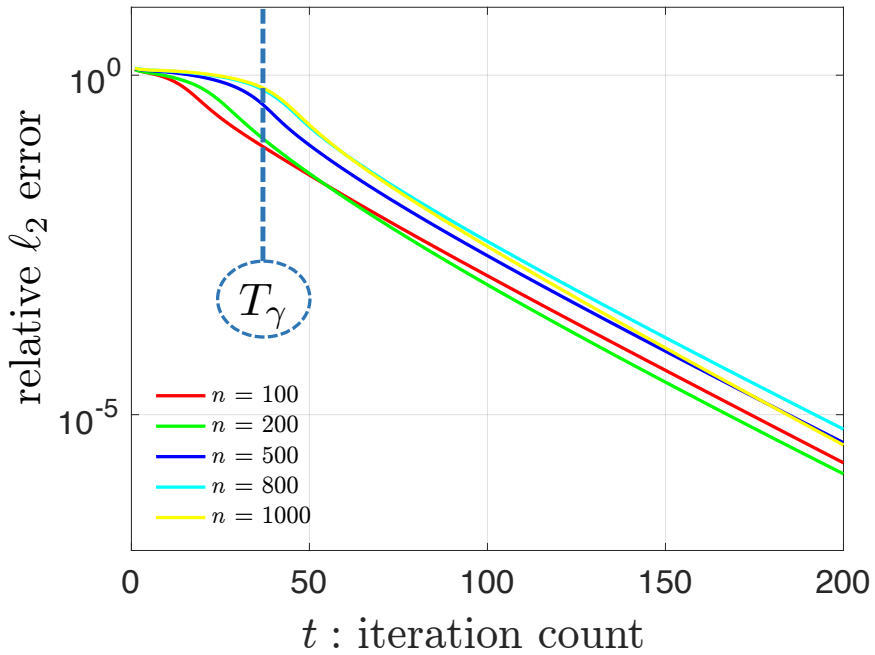
Under i.i.d. Gaussian design, GD with $\boldsymbol{x}^0 \sim \mathcal{N}(\boldsymbol{0}, n^{-1}\boldsymbol{I}_n)$ *achieves*

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \qquad t \ge T_\gamma$$

for $T_\gamma \lesssim \log n$ *and some constants* $\gamma, \rho > 0$*, provided that step size* $\eta \asymp 1$ *and sample size* $m \gtrsim n\,\mathrm{poly}\log m$

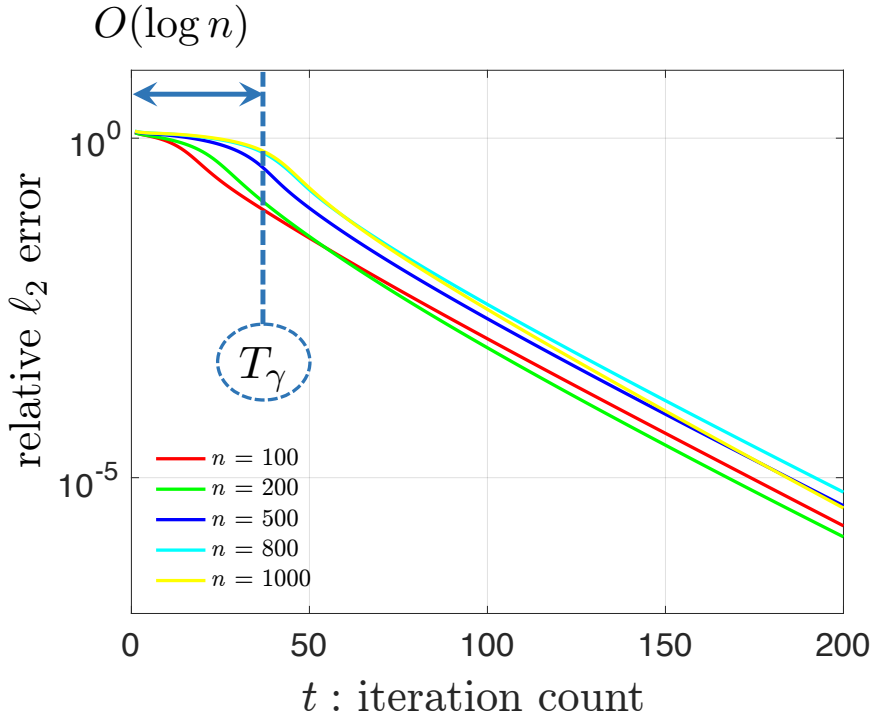
Theoretical guarantees for randomly initialized GD

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \leq \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \geq T_\gamma \asymp \log n$$



Theoretical guarantees for randomly initialized GD

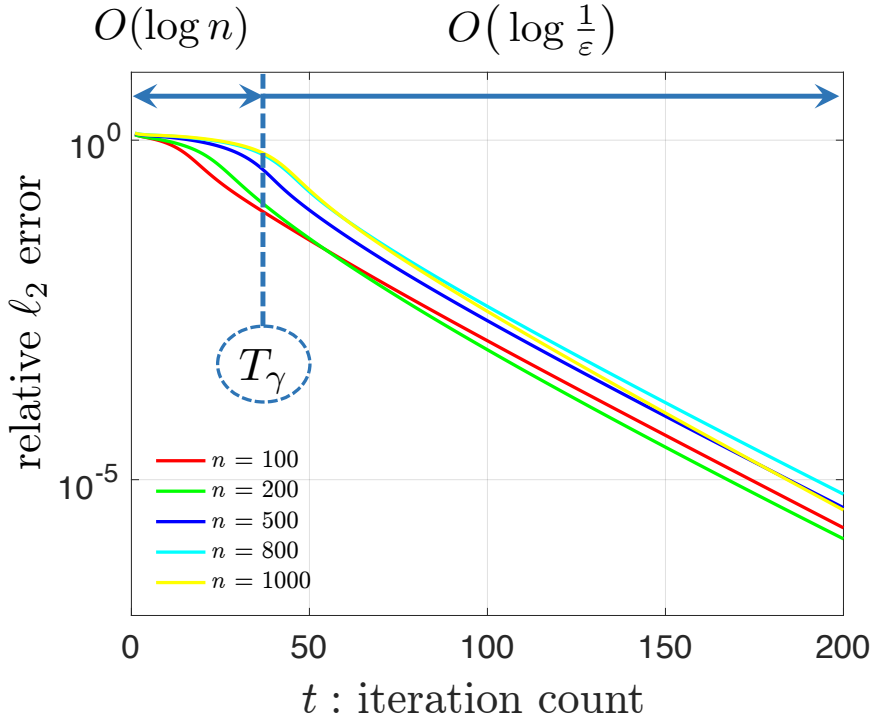
$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \leq \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \geq T_\gamma \asymp \log n$$



- *Stage 1:* takes $O(\log n)$ iterations to reach $\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \leq \gamma$

Theoretical guarantees for randomly initialized GD

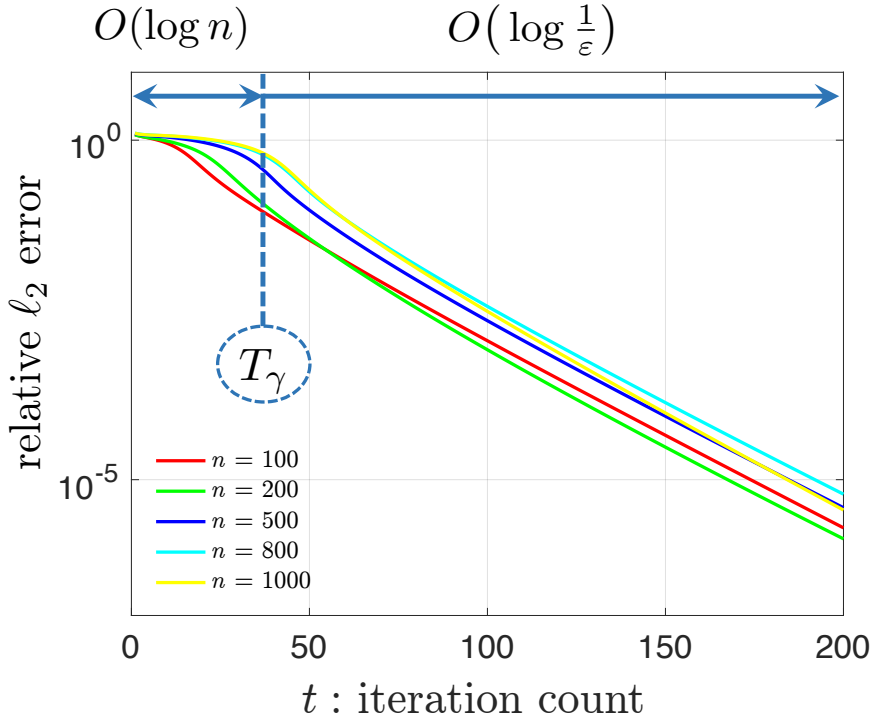
$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \ge T_\gamma \asymp \log n$$



- *Stage 1:* takes $O(\log n)$ iterations to reach $\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma$
- *Stage 2:* linear convergence

Theoretical guarantees for randomly initialized GD

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \ge T_\gamma \asymp \log n$$

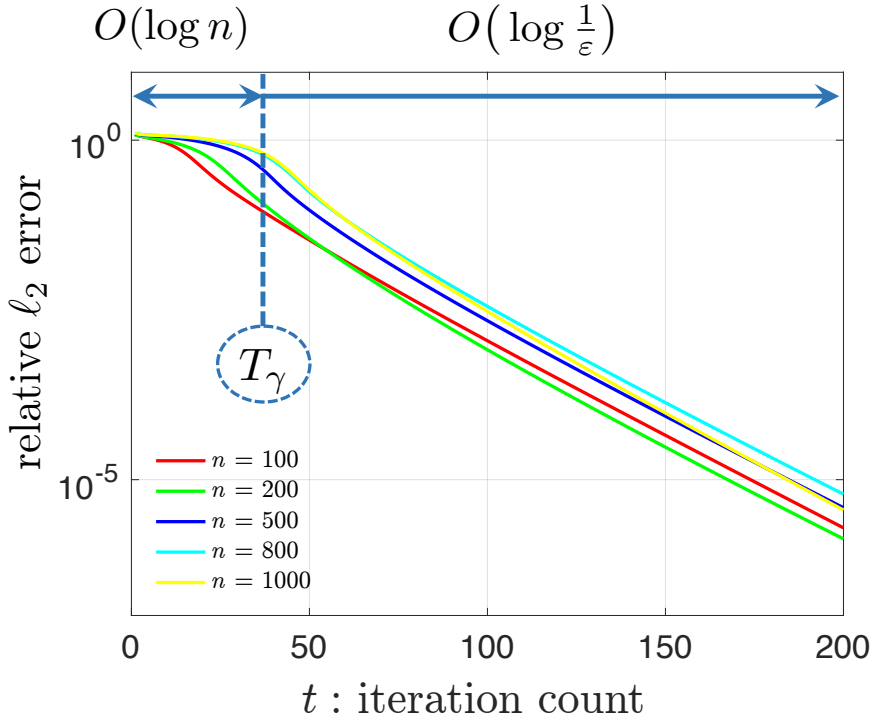


- *near-optimal compututational cost*:

  — $O\big(\log n + \log \frac{1}{\varepsilon}\big)$ iterations to yield ε accuracy

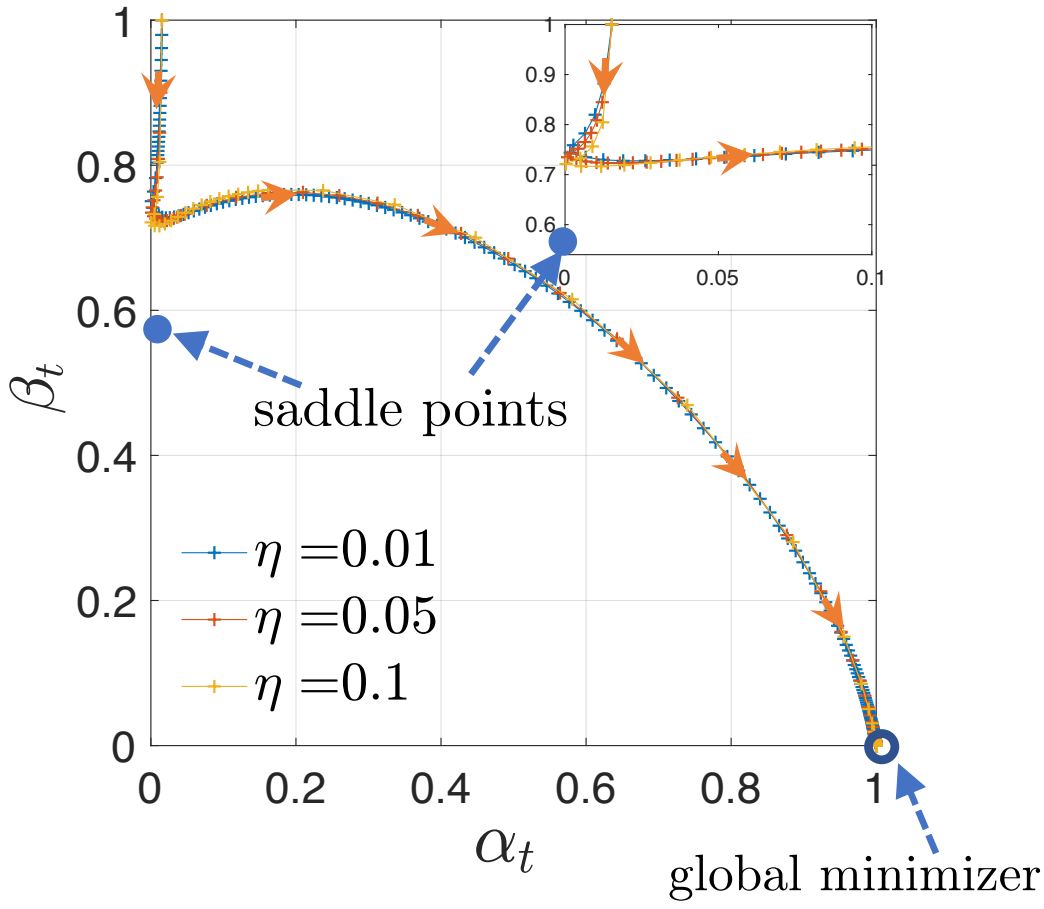
Theoretical guarantees for randomly initialized GD

$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \ge T_\gamma \asymp \log n$$



- *near-optimal compututational cost*:
  — $O\big(\log n + \log \frac{1}{\varepsilon}\big)$ iterations to yield ε accuracy
- *near-optimal sample size*: $m \gtrsim n\mathrm{poly}\log m$

Saddle-escaping schemes?



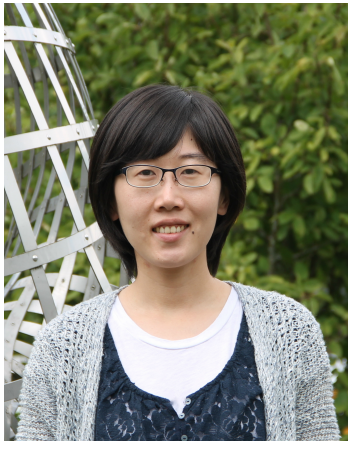
Randomly initialized GD never hits saddle points in phase retrieval!

Other saddle-escaping schemes

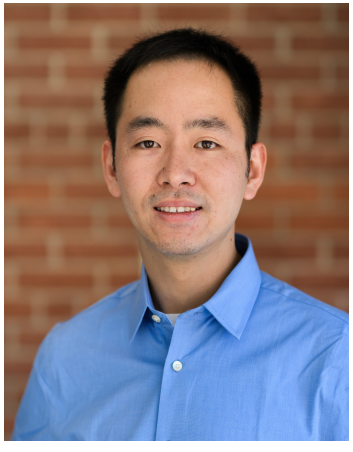
	iteration complexity	num of iterations needed to escape saddles	local iteration complexity
Trust-region (Sun et al. '16)	$n^7 + \log\log\frac{1}{\varepsilon}$	n^7	$\log\log\frac{1}{\varepsilon}$
Perturbed GD (Jin et al. '17)	$n^3 + n\log\frac{1}{\varepsilon}$	n^3	$n\log\frac{1}{\varepsilon}$
Perturbed accelerated GD (Jin et al. '17)	$n^{2.5} + \sqrt{n}\log\frac{1}{\varepsilon}$	$n^{2.5}$	$\sqrt{n}\log\frac{1}{\varepsilon}$
GD (Chen et al. '18)	$\log n + \log\frac{1}{\varepsilon}$	$\log n$	$\log\frac{1}{\varepsilon}$

Generic optimization theory yields highly suboptimal convergence guarantees

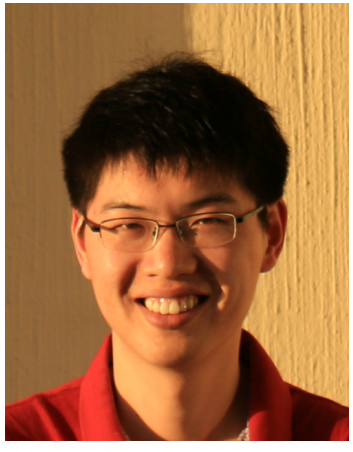
Nonconvex Optimization for High-Dimensional Signal Estimation: Spectral and Iterative Methods – Part III



Yuejie Chi
Carnegie Mellon



Yuxin Chen
Princeton



Cong Ma
UC Berkeley

EUSIPCO Tutorial, December 2020

Outline

- Part I: Introduction and Warm-Up

  Why nonconvex? basic concepts and a warm-up example (PCA)

- Part II: Gradient Descent and Implicit Regularization

  phase retrieval, matrix completion, random initialization

- Part III: Spectral Methods

  a general recipe, ℓ_2 and ℓ_∞ guarantees, community detection

- Part IV: Robustness to Corruptions and Ill-Conditioning

  median truncation, least absolute deviation, scaled gradient descent

Outline for Part III

- A motivating application: community detection
- A general recipe for spectral methods (with more applications)
- Classical spectral analysis: ℓ_2 perturbation theory
- Fine-grained analysis: ℓ_∞ perturbation theory
- A bird's-eye view of extensions

A motivating application: community detection

Community detection / graph clustering

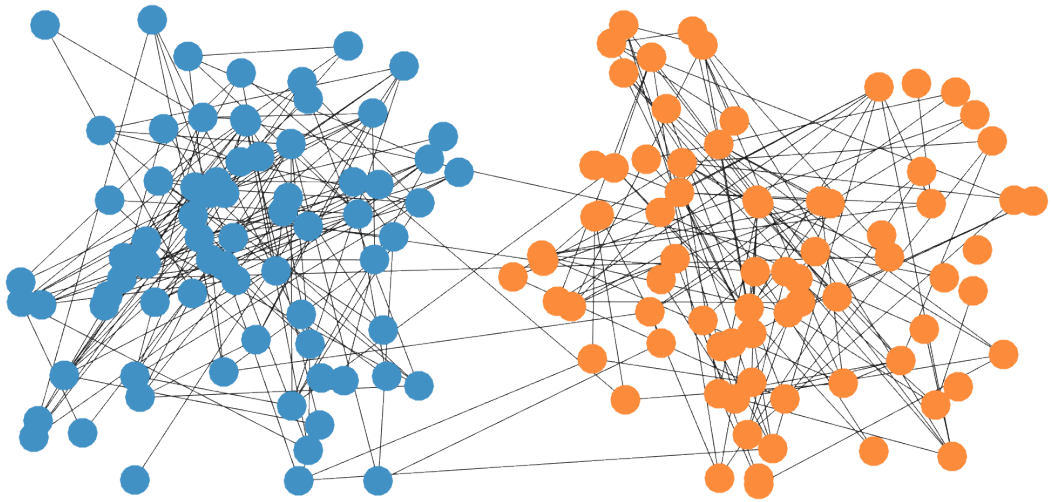
Community structures are common in many social networks

figure credit: The Future Buzz

figure credit: S. Papadopoulos

Goal: partition users into several clusters based on their friendships / similarities

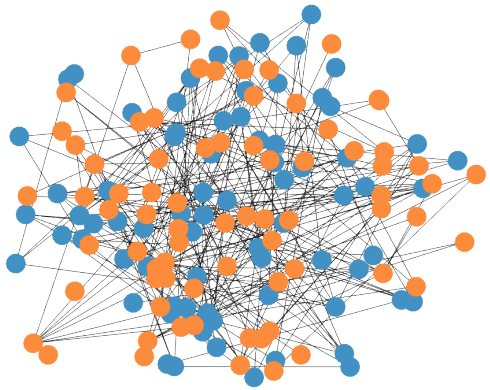
A simple model: stochastic block model (SBM)



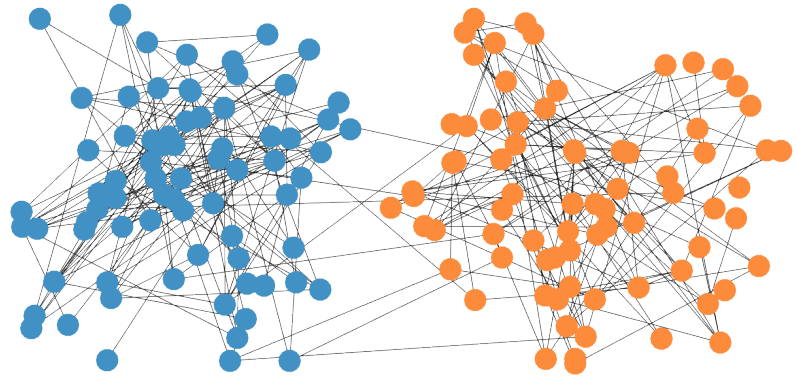
$x_i^\star = 1$: 1^{st} community $\qquad$ $x_i^\star = -1$: 2^{nd} community

- n nodes $\{1, \cdots, n\}$
- 2 communities
- n unknown variables: $x_1^\star, \cdots, x_n^\star \in \{1, -1\}$
  - encode community memberships

A simple model: stochastic block model (SBM)



$\mathcal{G}$



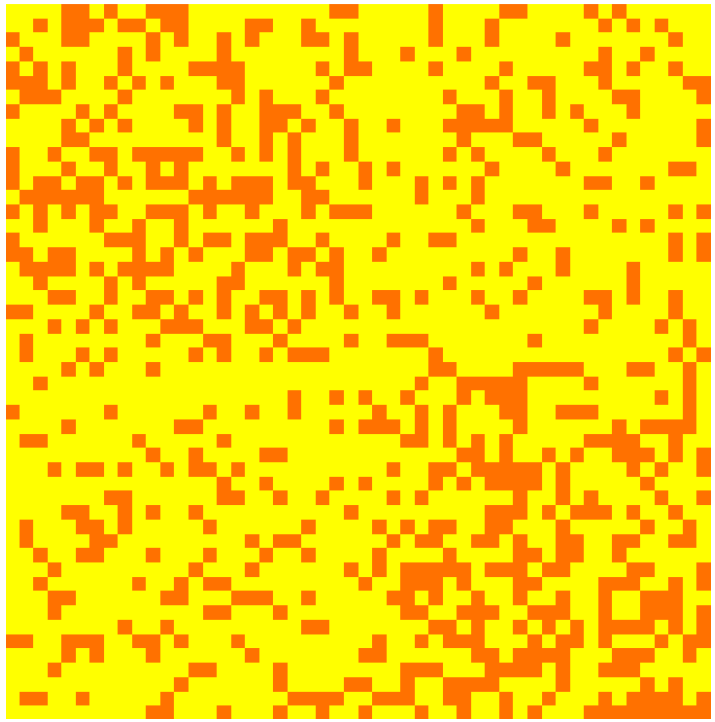
- observe a graph $\mathcal{G}$

  $(i,j) \in \mathcal{G}$ with prob. $\begin{cases} p, & \text{if } i \text{ and } j \text{ are from same community} \\ q, & \text{else} \end{cases}$

  Here, $p > q$

- **Goal:** recover community memberships of all nodes, i.e., $\{x_i^\star\}$

Adjacency matrix

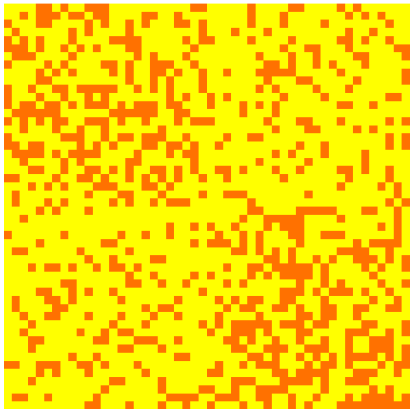
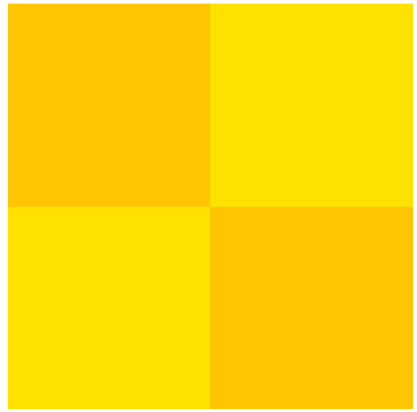
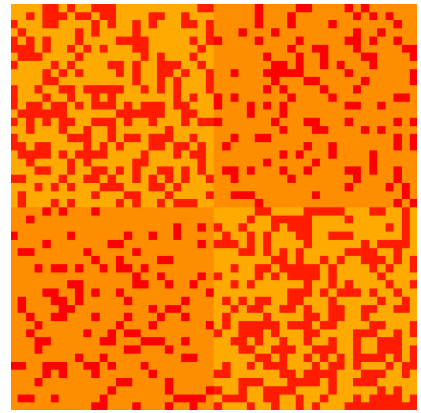


Consider the adjacency matrix $\boldsymbol{A} \in \{0,1\}^{n \times n}$ of $\mathcal{G}$:

$$A_{i,j} = \begin{cases} 1, & \text{if } (i,j) \in \mathcal{G} \\ 0, & \text{else} \end{cases}$$

- WLOG, suppose $x_1^\star = \cdots = x_{n/2}^\star = 1$; $x_{n/2+1}^\star = \cdots = x_n^\star = -1$

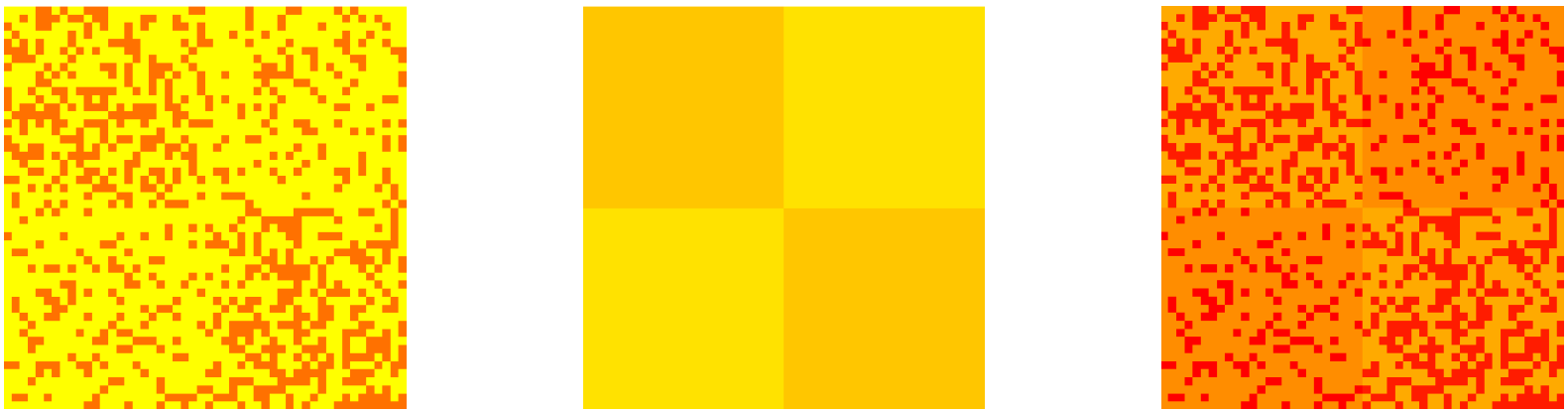
Adjacency matrix

$$\boldsymbol{A} = \underbrace{\mathbb{E}[\boldsymbol{A}]}_{\text{rank 2}} + \boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]$$

$$\mathbb{E}[\boldsymbol{A}] = \begin{bmatrix} p\mathbf{1}\mathbf{1}^\top & q\mathbf{1}\mathbf{1}^\top \\ q\mathbf{1}\mathbf{1}^\top & p\mathbf{1}\mathbf{1}^\top \end{bmatrix} = \underbrace{\frac{p+q}{2}\mathbf{1}\mathbf{1}^\top}_{\text{uninformative bias}} + \frac{p-q}{2} \underbrace{\begin{bmatrix} \mathbf{1} \\ -\mathbf{1} \end{bmatrix}}_{=\boldsymbol{x}^\star=[x_i]_{1\le i\le n}} [\mathbf{1}^\top, -\mathbf{1}^\top]$$

Spectral clustering



$$\boldsymbol{A} \quad = \quad \underbrace{\mathbb{E}[\boldsymbol{A}]}_{\text{rank 2}} \quad + \quad \boldsymbol{A} - \mathbb{E}\left[\boldsymbol{A}\right]$$

1. computing the leading eigenvector $\boldsymbol{u} = [u_i]_{1 \le i \le n}$ of $\boldsymbol{A} - \frac{p+q}{2}\mathbf{1}\mathbf{1}^\top$
2. rounding: output $x_i = \begin{cases} 1, & \text{if } u_i > 0 \\ -1, & \text{if } u_i < 0 \end{cases}$

Rationale behind spectral clustering

Recovery is reliable if $\underbrace{\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]}_{\text{perturbation}}$ is sufficiently small

- if $\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}] = \mathbf{0}$, then

$$\boldsymbol{u} \propto \pm \begin{bmatrix} \mathbf{1} \\ -\mathbf{1} \end{bmatrix} \implies \text{perfect clustering}$$

A general recipe for spectral methods

Three key steps:

- identify a key matrix $\boldsymbol{M}^\star$, whose eigenvectors disclose crucial information
- construct a surrogate matrix $\boldsymbol{M}$ of $\boldsymbol{M}^\star$ using data
- compute corresponding eigenvectors of $\boldsymbol{M}$

Low-rank matrix completion

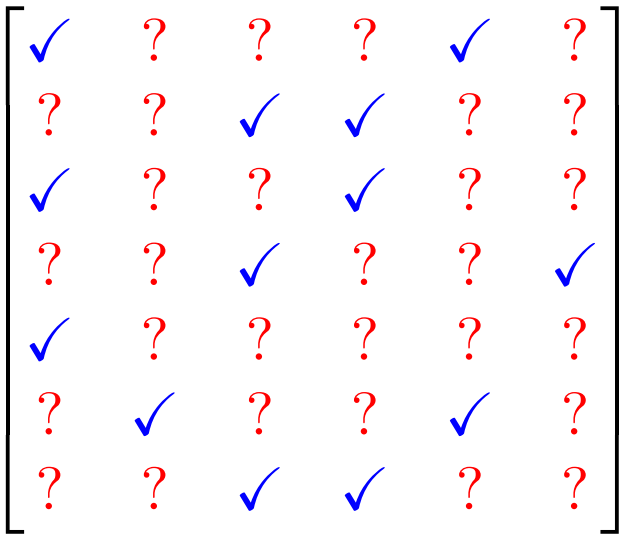


figure credit: Candès

- consider a low-rank matrix $\boldsymbol{M}^\star = \boldsymbol{U}^\star \boldsymbol{\Sigma}^\star \boldsymbol{V}^{\star\top}$
- each entry $M^\star_{i,j}$ is observed independently with prob. p
- **intermediate goal:** estimate $\boldsymbol{U}^\star, \boldsymbol{V}^\star$

Spectral method for matrix completion

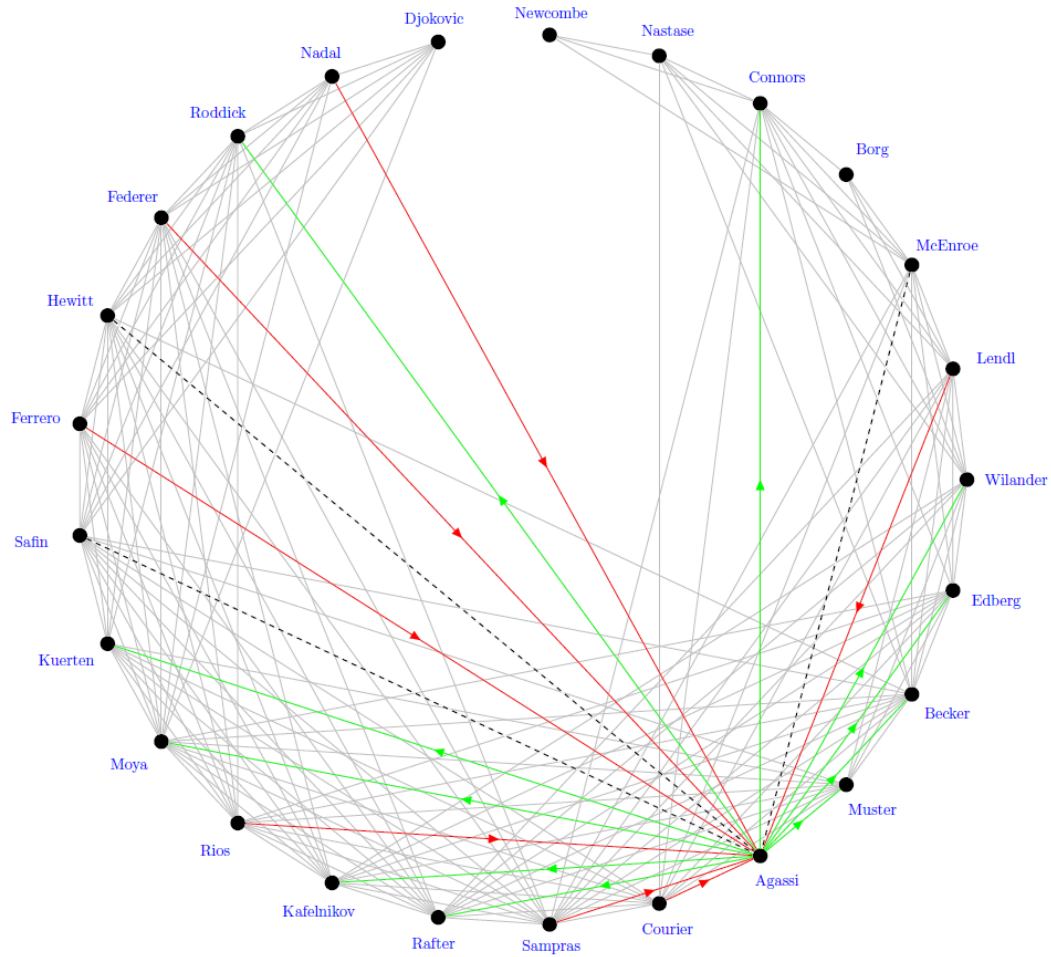
1. identify the key matrix $\boldsymbol{M}^\star$
2. construct surrogate matrix $\boldsymbol{M} \in \mathbb{R}^{n \times n}$ as

$$M_{i,j} = \begin{cases} \frac{1}{p} M^\star_{i,j}, & \text{if } M^\star_{i,j} \text{ is observed} \\ 0, & \text{else} \end{cases}$$

   - **rationale for rescaling:** ensures $\mathbb{E}[\boldsymbol{M}] = \boldsymbol{M}^\star$

3. compute the rank-r SVD $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ of $\boldsymbol{M}$, and return $(\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V})$

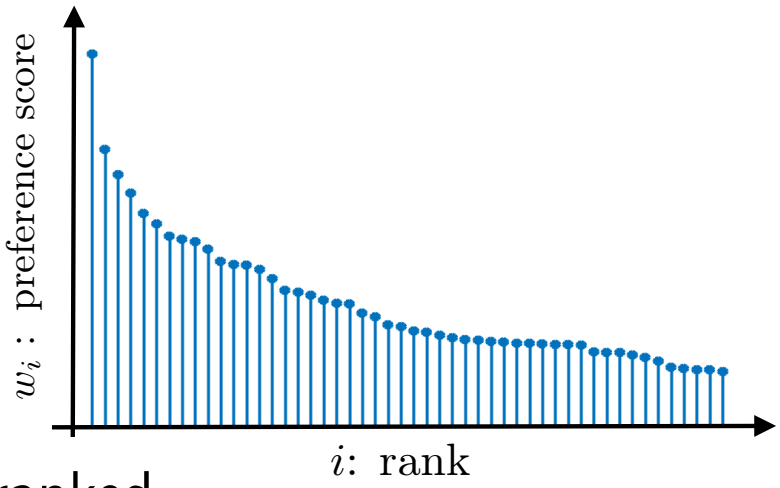
Ranking from pairwise comparisons



pairwise comparisons for ranking tennis players

figure credit: Bozóki, Csató, Temesi

Bradley-Terry-Luce (logistic) model



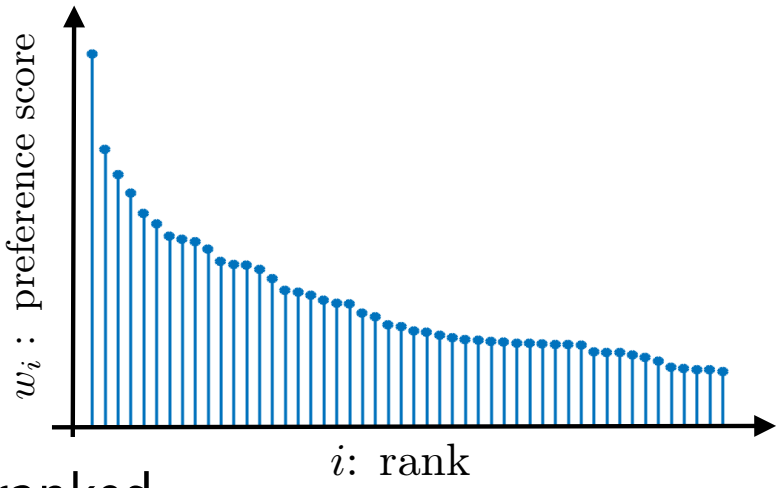
- n items to be ranked
- assign a latent score $\{w_i^\star\}_{1\le i\le n}$ to each item, so that

$$\text{item } i \succ \text{item } j \quad \text{if} \quad w_i^\star > w_j^\star$$

- each pair of items (i, j) is compared independently

$$\mathbb{P}\{\text{item } j \text{ beats item } i\} = \frac{w_j^\star}{w_i^\star + w_j^\star}$$

Bradley-Terry-Luce (logistic) model



- n items to be ranked
- assign a latent score $\{w_i^\star\}_{1\le i\le n}$ to each item, so that

$$\text{item } i \succ \text{item } j \quad \text{if} \quad w_i^\star > w_j^\star$$

- each pair of items (i,j) is compared independently

$$y_{i,j} \overset{\text{ind.}}{=} \begin{cases} 1, & \text{with prob. } \frac{w_j^\star}{w_i^\star + w_j^\star} \\ 0, & \text{else} \end{cases}$$

- **intermediate goal:** estimate score vector $\boldsymbol{w}^\star$ (up to scaling)

Spectral ranking

1. identify key matrix $\boldsymbol{P}^\star$—probability transition matrix

$$P^\star_{i,j} = \begin{cases} \frac{1}{n} \cdot \frac{w^\star_j}{w^\star_i + w^\star_j}, & \text{if } i \neq j \\ 1 - \sum_{l:l\neq i} P^\star_{i,l}, & \text{if } i = j \end{cases}$$

Rationale:

- $\boldsymbol{P}^\star$ obeys

$$w^\star_i P^\star_{i,j} = w^\star_j P^\star_{j,i} \qquad \text{(detailed balance)}$$

- Thus, the stationary distribution $\boldsymbol{\pi}^\star$ of $\boldsymbol{P}^\star$ obeys

$$\boldsymbol{\pi}^\star \;=\; \frac{1}{\sum_l w^\star_l} \boldsymbol{w}^\star \qquad \text{(reveals true scores)}$$

Spectral ranking

2. construct a surrogate matrix $\boldsymbol{P}$ obeying

$$P_{i,j} = \begin{cases} \frac{1}{n} y_{i,j}, & \text{if } i \neq j \\ 1 - \sum_{l:l \neq i} P_{i,l}, & \text{if } i = j \end{cases}$$

3. return leading left eigenvector $\boldsymbol{\pi}$ of $\boldsymbol{P}$ as score estimate

— closely related to PageRank

Spectral ranking

2. construct a surrogate matrix $\boldsymbol{P}$ obeying

$$P_{i,j} = \begin{cases} \frac{1}{n} y_{i,j}, & \text{if } i \neq j \\ 1 - \sum_{l:l \neq i} P_{i,l}, & \text{if } i = j \end{cases}$$

3. return leading left eigenvector $\boldsymbol{\pi}$ of $\boldsymbol{P}$ as score estimate

— closely related to PageRank

Key: stability of eigenspace against perturbation $\boldsymbol{M} - \boldsymbol{M}^\star$?

Classical spectral analysis: ℓ_2 perturbation theory

Setup and notation

Consider two symmetric matrices $\boldsymbol{M}^\star$ and its perturbed version

$$\boldsymbol{M} = \boldsymbol{M}^\star + \boldsymbol{E} \in \mathbb{R}^{n \times n}$$

with eigendecompositions

$$\boldsymbol{M}^\star = \sum_{i=1}^{n} \lambda_i^\star \boldsymbol{u}_i^\star \boldsymbol{u}_i^{\star\top} = \begin{bmatrix} \boldsymbol{U}^\star & \boldsymbol{U}_\perp^\star \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}^\star & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Lambda}_\perp^\star \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^{\star\top} \\ \boldsymbol{U}_\perp^{\star\top} \end{bmatrix};$$

$$\boldsymbol{M} = \sum_{i=1}^{n} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^\top = \begin{bmatrix} \boldsymbol{U} & \boldsymbol{U}_\perp \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Lambda}_\perp \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^\top \\ \boldsymbol{U}_\perp^\top \end{bmatrix}$$

Setup and notation

$$\boldsymbol{M}^\star = \underbrace{\begin{bmatrix} \boldsymbol{u}_1^\star & \cdots & \boldsymbol{u}_r^\star \end{bmatrix}}_{\boldsymbol{U}^\star} \underbrace{\begin{bmatrix} \boldsymbol{u}_{r+1}^\star & \cdots & \boldsymbol{u}_n^\star \end{bmatrix}}_{\boldsymbol{U}_\perp^\star} \cdot \begin{bmatrix} \underbrace{\begin{bmatrix} \lambda_1^\star & & \\ & \ddots & \\ & & \lambda_r^\star \end{bmatrix}}_{\boldsymbol{\Lambda}^\star} & \\ & \underbrace{\begin{bmatrix} \lambda_{r+1}^\star & & \\ & \ddots & \\ & & \lambda_n^\star \end{bmatrix}}_{\boldsymbol{\Lambda}_\perp^\star} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{u}_1^{\star\top} \\ \vdots \\ \boldsymbol{u}_r^{\star\top} \\ \boldsymbol{u}_{r+1}^{\star\top} \\ \vdots \\ \boldsymbol{u}_n^{\star\top} \end{bmatrix} \begin{matrix} \} \boldsymbol{U}^{\star\top} \\ \\ \} \boldsymbol{U}_\perp^{\star\top} \end{matrix}$$

Eigenspace perturbation theory

Main focus: how does the perturbation $\boldsymbol{E}$ affect the distance between $\boldsymbol{U}$ and $\boldsymbol{U}^\star$?

Question: how to define distance between two subspaces?

- $\|\boldsymbol{U} - \boldsymbol{U}^\star\|_{\mathrm{F}}$ and $\|\boldsymbol{U} - \boldsymbol{U}^\star\|$ are not appropriate, since they fall short of accounting for $\underbrace{\text{global orthonormal transformation}}$

  $\forall$ orthonormal $\boldsymbol{R} \in \mathbb{R}^{r\times r}$, $\boldsymbol{U}$ and $\boldsymbol{U}\boldsymbol{R}$ represent same subspace

Distance between two subspaces

One solution: taking best rotation into consideration

$$\mathsf{dist}(\boldsymbol{U}, \boldsymbol{U}^\star) := \min_{\boldsymbol{R} \in \mathcal{O}^{r \times r}} \left\| \boldsymbol{U}\boldsymbol{R} - \boldsymbol{U}^\star \right\|;$$

$$\mathsf{dist}_{\mathrm{F}}(\boldsymbol{U}, \boldsymbol{U}^\star) := \min_{\boldsymbol{R} \in \mathcal{O}^{r \times r}} \left\| \boldsymbol{U}\boldsymbol{R} - \boldsymbol{U}^\star \right\|_{\mathrm{F}}$$

Davis-Kahan $\sin\Theta$ Theorem: a simple case

Chandler Davis

William Kahan

Theorem 1

Suppose $\boldsymbol{M}^\star \succeq \mathbf{0}$ *and has rank* r*. If* $\|\boldsymbol{E}\| < \lambda_r(\boldsymbol{M}^\star)$*, then*

$$\operatorname{dist}\big(\boldsymbol{U}, \boldsymbol{U}^\star\big) \leq \frac{\big\|\boldsymbol{E}\boldsymbol{U}^\star\big\|}{\lambda_r(\boldsymbol{M}^\star) - \|\boldsymbol{E}\|} \leq \frac{\|\boldsymbol{E}\|}{\lambda_r(\boldsymbol{M}^\star) - \|\boldsymbol{E}\|}$$

- depends on $\underbrace{\text{smallest non-zero eigenvalue of } \boldsymbol{M}^\star}_{\text{eigengap between } \lambda_r(\boldsymbol{M}^\star) \text{ and } \lambda_{r+1}(\boldsymbol{M}^\star)}$ and perturbation size

Back to stochastic block model

Let $\boldsymbol{M}^\star := \mathbb{E}[\boldsymbol{A}] - \frac{p+q}{2}\mathbf{1}\mathbf{1}^\top = \frac{p-q}{2}\begin{bmatrix} \mathbf{1} \\ -\mathbf{1} \end{bmatrix}\begin{bmatrix} \mathbf{1}^\top & -\mathbf{1}^\top \end{bmatrix}$,

$\boldsymbol{M} := \boldsymbol{A} - \frac{p+q}{2}\mathbf{1}\mathbf{1}^\top$, and $\boldsymbol{u}^\star := \frac{1}{\sqrt{n}}\begin{bmatrix} \mathbf{1} \\ -\mathbf{1} \end{bmatrix}$

Then the Davis-Kahan $\sin\boldsymbol{\Theta}$ Theorem yields

$$\mathsf{dist}(\boldsymbol{u}, \boldsymbol{u}^\star) \leq \frac{\|\boldsymbol{M} - \boldsymbol{M}^\star\|}{\lambda_1(\boldsymbol{M}^\star) - \|\boldsymbol{M} - \boldsymbol{M}\|} = \frac{\|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\|}{\frac{(p-q)n}{2} - \|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\|} \quad (1)$$

as long as $\|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\| < \lambda_1(\boldsymbol{M}^\star) = \frac{(p-q)n}{2}$

Bounding $\|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\|$

Matrix concentration inequalities tell us that

Lemma 2

Consider SBM with $p > q$ and $p \gtrsim \frac{\log n}{n}$. Then with high prob.

$$\|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\| \lesssim \sqrt{np} \tag{2}$$

Statistical accuracy of spectral clustering

Substitute (2) into (1) to reach

$$\text{dist}(\boldsymbol{u}, \boldsymbol{u}^\star) \le \frac{\|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\|}{\frac{(p-q)n}{2} - \|\boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]\|} \lesssim \frac{\sqrt{np}}{(p-q)n}$$

provided that $(p-q)n \gg \sqrt{np}$

Thus, under condition $\frac{p-q}{\sqrt{p}} \gg \sqrt{\frac{1}{n}}$, with high prob. one has

$$\text{dist}(\boldsymbol{u}, \boldsymbol{u}^\star) \ll 1 \qquad \Longrightarrow \qquad \text{nearly perfect clustering}$$

Statistical accuracy of spectral clustering

$$\frac{p-q}{\sqrt{p}} \gg \sqrt{\frac{1}{n}} \quad \Longrightarrow \quad \text{nearly perfect clustering}$$

- **dense regime:** if $p \asymp q \asymp 1$, then this condition reads

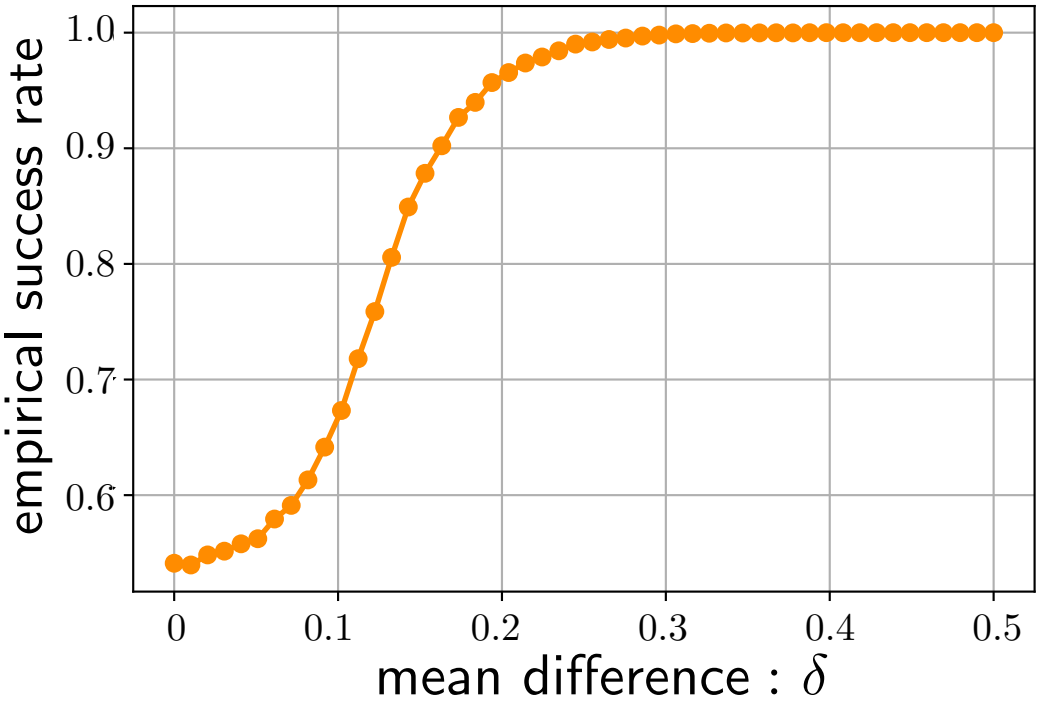
$$p - q \gg \sqrt{\frac{1}{n}}$$

- **"sparse" regime:** if $p = \frac{a \log n}{n}$ and $q = \frac{b \log n}{n}$ for $a, b \asymp 1$, then

$$a - b \gg \sqrt{a \log n}$$

This condition is information-theoretically optimal (up to log factor)
— *Mossel, Neeman, Sly '15, Abbe '18*

Empirical performance of spectral clustering



ℓ_2 perturbation theory alone cannot explain exact recovery guarantees

— call for fine-grained analysis

Reverse engineering

Spectral clustering uses signs of $\boldsymbol{u}$ to cluster nodes

Reverse engineering

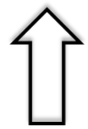
Spectral clustering uses signs of $\boldsymbol{u}$ to cluster nodes

It achieves exact recovery iff $u_i u_i^\star > 0$ for all i

Reverse engineering

Spectral clustering uses signs of $\boldsymbol{u}$ to cluster nodes

It achieves exact recovery iff $u_i u_i^\star > 0$ for all i

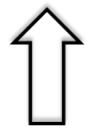


A sufficient condition is* $\|\boldsymbol{u} - \boldsymbol{u}^\star\|_\infty < 1/\sqrt{n}$

Reverse engineering

Spectral clustering uses signs of $\boldsymbol{u}$ to cluster nodes

It achieves exact recovery iff $u_i u_i^\star > 0$ for all i



A sufficient condition is* $\|\boldsymbol{u} - \boldsymbol{u}^\star\|_\infty < 1/\sqrt{n}$

Need ℓ_∞ perturbation theory

Fine-grained analysis: ℓ_∞ perturbation theory

Setup and notation (rank-1 case)

Groundtruth: consider a rank-1 psd matrix $\boldsymbol{M}^\star = \lambda^\star \boldsymbol{u}^\star \boldsymbol{u}^{\star\top} \in \mathbb{R}^{n\times n}$

Incoherence: define

$$\mu := n\|\boldsymbol{u}^\star\|_\infty^2 \qquad (1 \le \mu \le n)$$

Observations:

$$\boldsymbol{M} = \boldsymbol{M}^\star + \boldsymbol{E} \in \mathbb{R}^{n\times n}$$

with $\boldsymbol{E}$ a symmetric noise matrix

Noise assumptions

The entries in the lower triangular part of $\boldsymbol{E} = [E_{i,j}]_{1 \le i,j \le n}$ are independently generated obeying

$$\mathbb{E}[E_{i,j}] = 0, \quad \mathbb{E}[E_{i,j}^2] \le \sigma^2, \quad |E_{i,j}| \le B, \quad \text{for all } i \ge j$$

Further, assume that

$$c_{\mathsf{b}} \coloneqq \frac{B}{\sigma \sqrt{n/(\mu \log n)}} = O(1)$$

ℓ_∞ perturbation theory

Theorem 3

With high prob, there exists $z \in \{1, -1\}$ *such that*

$$\|z\boldsymbol{u} - \boldsymbol{u}^\star\|_\infty \lesssim \frac{\sigma\sqrt{\mu} + \sigma\sqrt{\log n}}{\lambda^\star}, \tag{3a}$$

$$\|z\boldsymbol{u} - \frac{1}{\lambda^\star}\boldsymbol{M}\boldsymbol{u}^\star\|_\infty \lesssim \frac{\sigma\sqrt{\mu}}{\lambda^\star} + \frac{\sigma^2\sqrt{n\log n} + \sigma B\sqrt{\mu\log^3 n}}{(\lambda^\star)^2} \tag{3b}$$

provided that $\sigma\sqrt{n\log n} \le c_\sigma\lambda^\star$ *for some sufficiently small constant* $c_\sigma > 0$.

Key message:

- when $\mu \lesssim \sqrt{\log n}$, (3a) is $\sqrt{n/\log n}$ smaller than ℓ_2 bound

$$\|z\boldsymbol{u} - \boldsymbol{u}^\star\|_2 \lesssim \sigma\sqrt{n}/\lambda^\star$$

Back to stochastic block model

Recall $\boldsymbol{M}^\star := \mathbb{E}[\boldsymbol{A}] - \frac{p+q}{2}\mathbf{1}\mathbf{1}^\top = \frac{p-q}{2}\begin{bmatrix} \mathbf{1} \\ -\mathbf{1} \end{bmatrix}\begin{bmatrix} \mathbf{1}^\top & -\mathbf{1}^\top \end{bmatrix}$,

$\boldsymbol{M} := \boldsymbol{A} - \frac{p+q}{2}\mathbf{1}\mathbf{1}^\top$, and $\boldsymbol{u}^\star := \frac{1}{\sqrt{n}}\begin{bmatrix} \mathbf{1} \\ -\mathbf{1} \end{bmatrix}$

These imply

$$\begin{aligned} \lambda^\star &= \tfrac{n(p-q)}{2} \\ \mu &= 1 \\ B &= 1 \\ \sigma^2 &\le \max\{p, q\} = p \end{aligned}$$

Invoke ℓ_∞ perturbation theory

ℓ_∞ perturbation theory (3b) yields

$$\begin{aligned}\left\|z\lambda^\star \boldsymbol{u} - \boldsymbol{M}\boldsymbol{u}^\star\right\|_\infty &\lesssim \sigma + \frac{\sigma^2\sqrt{n\log n}}{\lambda^\star} + \frac{\sigma B\,\log^{3/2} n}{\lambda^\star} \\ &\le C\Big(\sqrt{p} + \frac{p\sqrt{\log n}}{\sqrt{n}(p-q)} + \frac{\sqrt{p}\log^{3/2} n}{n(p-q)}\Big) =: \Delta\end{aligned}$$

for some constant $C > 0$

it boils down to controlling the entrywise behavior of $\boldsymbol{M}\boldsymbol{u}^\star$

Bounding entries in $M u^\star$

Again concentration inequalities tell us that

Lemma 4

Suppose that

$$\left(\sqrt{p} - \sqrt{q}\right)^2 \geq (1+\varepsilon)\,\frac{2\log n}{n} \tag{4}$$

for some quantity $\varepsilon > 0$. *Let* $\varepsilon_0 := \frac{\varepsilon \log n}{\sqrt{n}\log\frac{p(1-q)}{q(1-p)}} - \frac{1}{\sqrt{n}}$. *Then with probability exceeding* $1 - n^{-\varepsilon/2}$, *one has*

$$\boldsymbol{M}_{l,\cdot}\boldsymbol{u}^\star \geq \varepsilon_0 \;\text{ for all } l \leq \frac{n}{2} \quad \text{and} \quad \boldsymbol{M}_{l,\cdot}\boldsymbol{u}^\star \leq -\varepsilon_0 \;\text{ for all } l > \frac{n}{2}.$$

Key message: entries in $\boldsymbol{M}\boldsymbol{u}^\star$ are bounded away from 0 with correct sign

Completing the picture

On one hand

$$\boldsymbol{M}_{l,\cdot}\boldsymbol{u}^\star \geq \varepsilon_0 \text{ for all } l \leq \frac{n}{2} \quad \text{and} \quad \boldsymbol{M}_{l,\cdot}\boldsymbol{u}^\star \leq -\varepsilon_0 \text{ for all } l > \frac{n}{2}.$$

On the other hand

$$\left\| z\lambda^\star \boldsymbol{u} - \boldsymbol{M}\boldsymbol{u}^\star \right\|_\infty \leq \Delta$$

In sum, if one can show

$$\varepsilon_0 > \Delta$$

then it follows that

$$z u_l u_l^\star > 0 \quad \text{for all } 1 \leq l \leq n \quad \Longrightarrow \quad \text{exact recovery}$$

Exact recovery of SBM

Theorem 5

Fix any constant $\varepsilon > 0$. *Suppose* $p = \frac{\alpha \log n}{n}$ *and* $q = \frac{\beta \log n}{n}$ *for some sufficiently large constants* $\alpha > \beta > 0$. *In addition, assume that*

$$\left(\sqrt{p} - \sqrt{q}\right)^2 \geq 2\left(1 + \varepsilon\right) \frac{\log n}{n}. \tag{5}$$

With probability $1 - o(1)$, *spectral clustering achieves exact recovery.*

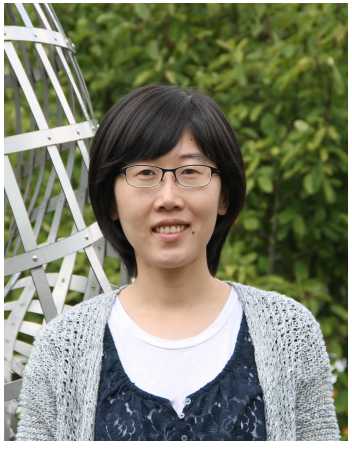
This condition is information-theoretically optimal

A bird's-eye view of extensions

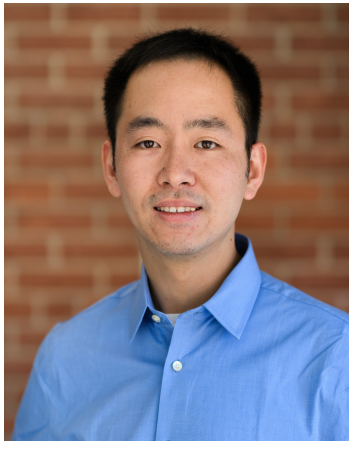
- Davis-Kahan for general symmetric matrices (not necessarily PSD)
- Wedin's theorem on singular subspace perturbation theory
- Eigenvector perturbation for probability transition matrices
- General $\ell_{2,\infty}$ eigenspace and singular space perturbation

Advertisement: "Spectral Methods for Data Science: A Statistical Perspective", Y. Chen, Y. Chi, J. Fan and C. Ma, 2020

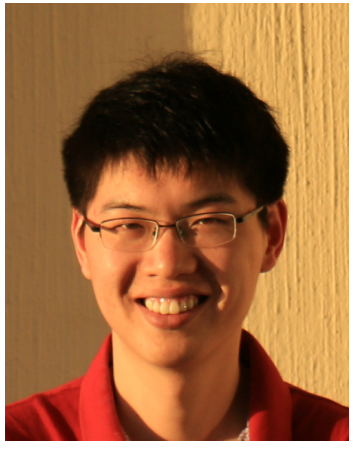
Nonconvex Optimization for High-Dimensional Signal Estimation: Spectral and Iterative Methods – Part IV



Yuejie Chi
Carnegie Mellon



Yuxin Chen
Princeton



Cong Ma
UC Berkeley

EUSIPCO Tutorial, December 2020

Outline

- Part I: Introduction and Warm-Up

  Why nonconvex? basic concepts and a warm-up example (PCA)

- Part II: Gradient Descent and Implicit Regularization

  phase retrieval, matrix completion, random initialization

- Part III: Spectral Methods

  a general recipe, ℓ_2 and ℓ_∞ guarantees, community detection

- Part IV: Robustness to Corruptions and Ill-Conditioning

  median truncation, least absolute deviation, scaled gradient descent

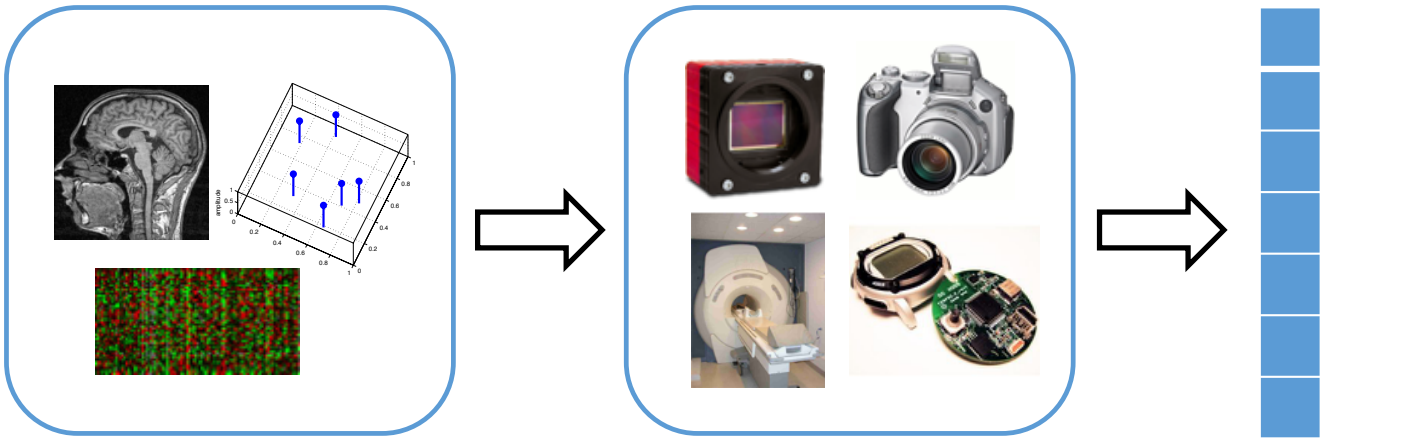
Robustness to ill-conditioning?

A factorization approach to low-rank matrix sensing

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

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

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

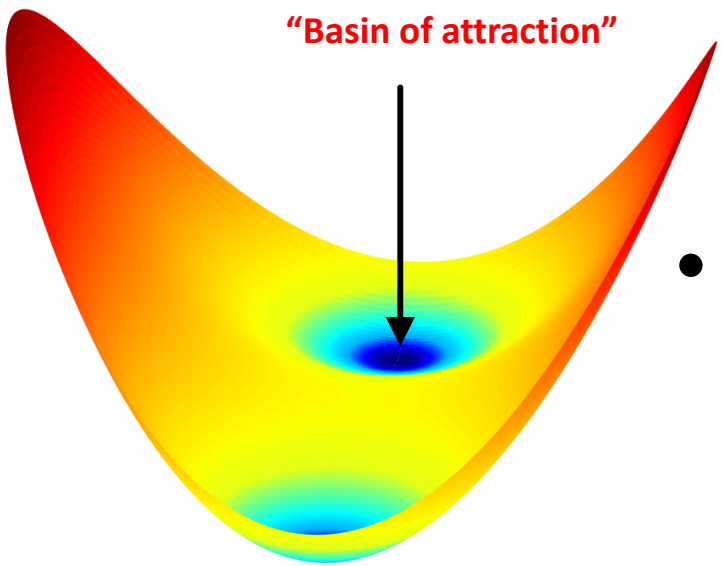


$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{M}) + \text{noise}$$

find $\boldsymbol{X} \in \mathbb{R}^{n_1 \times r}$, $\boldsymbol{Y} = \mathbb{R}^{n_2 \times r}$, such that $\boldsymbol{y} \approx \mathcal{A}(\boldsymbol{X}\boldsymbol{Y}^\top)$

Prior art: GD with balancing regularization

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



- **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:**

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

for $t = 0, 1, \ldots$

Prior theory for vanilla GD

Theorem 1 (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

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

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

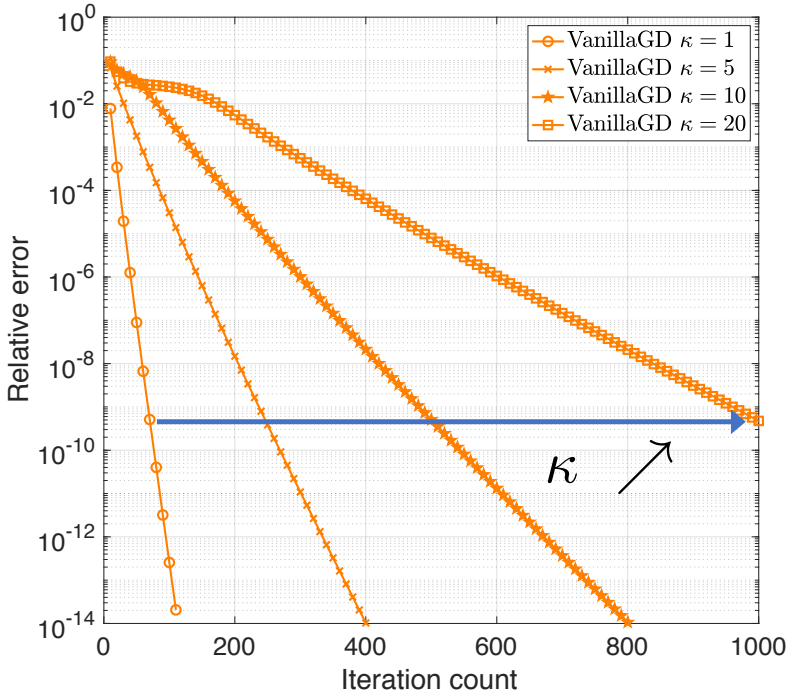
$$m \gtrsim (n_1 + n_2) r^2 \kappa^2.$$

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

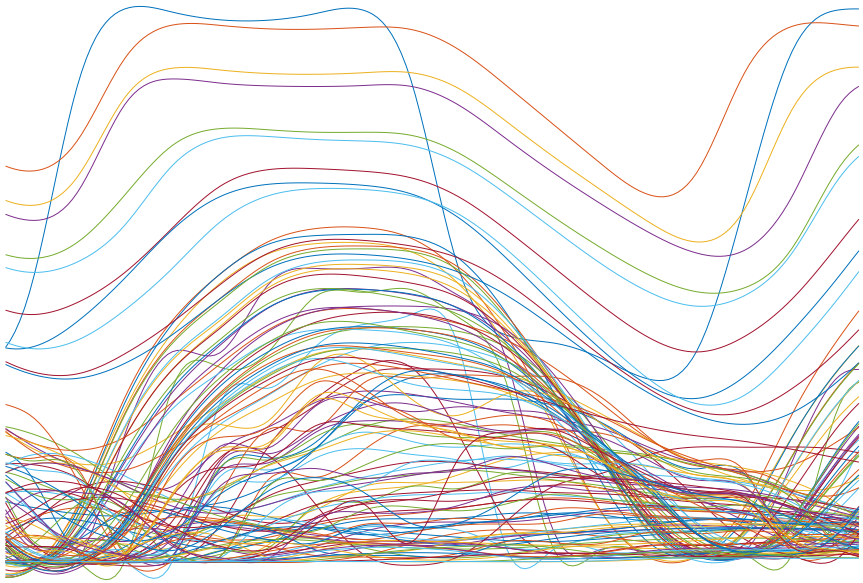
$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad f(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\left\|\mathcal{P}_{\Omega}(\boldsymbol{X}\boldsymbol{Y}^{\top} - \boldsymbol{M})\right\|_{\mathrm{F}}^{2}$$



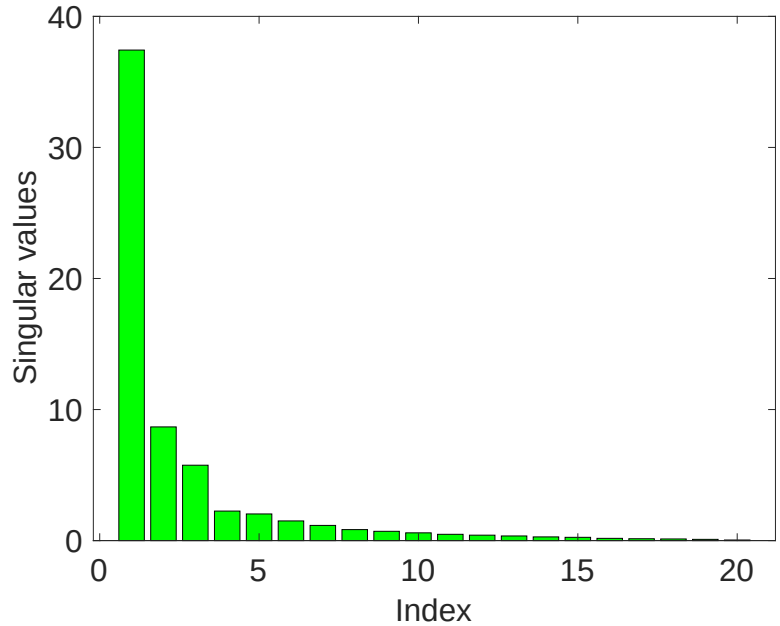
Vanilla GD converges in $O\big(\kappa \log \frac{1}{\varepsilon}\big)$ iterations.

— *Can we provably accelerate the convergence to* $O\big(\log \frac{1}{\varepsilon}\big)$*?*

Condition number can be large



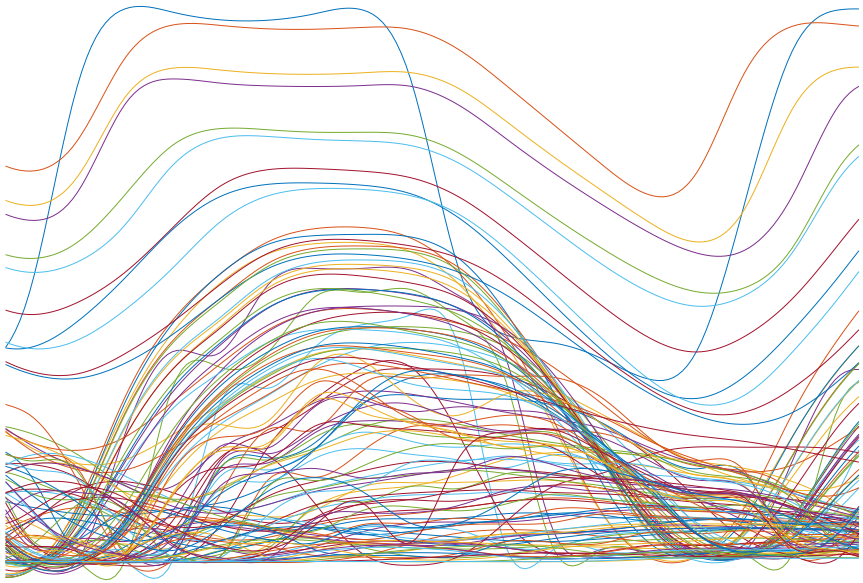
chlorine concentration levels
120 junctions, 180 time slots



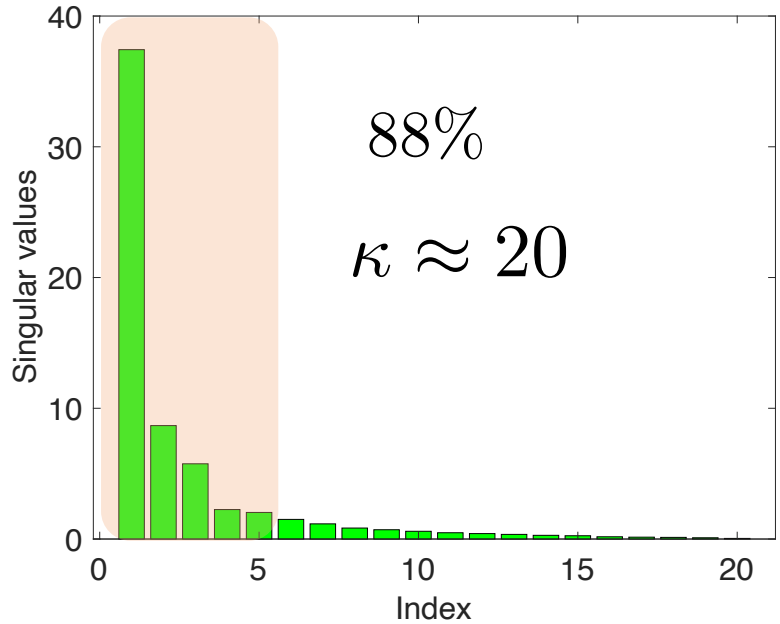
power-law spectrum

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

Condition number can be large



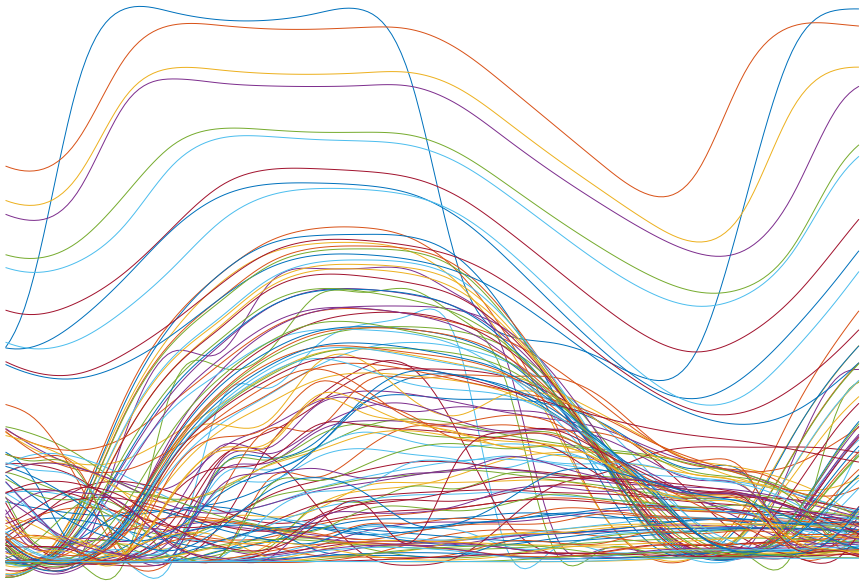
chlorine concentration levels
120 junctions, 180 time slots



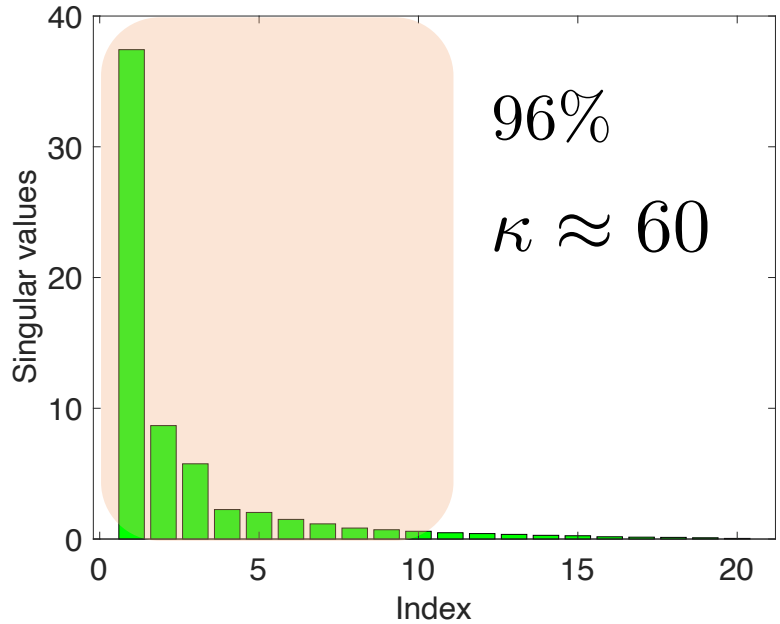
rank-5 approximation

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

Condition number can be large



chlorine concentration levels
120 junctions, 180 time slots

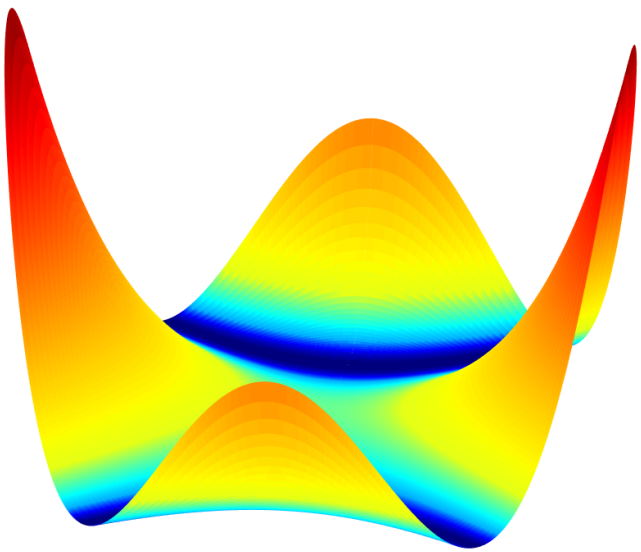


rank-10 approximation

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

A new algorithm: scaled gradient descent

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



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

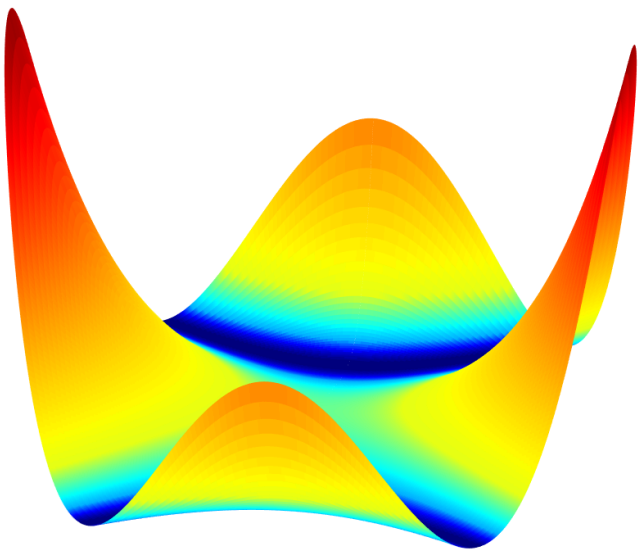
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

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

for $t = 0, 1, \ldots$

A new algorithm: scaled gradient descent

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



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

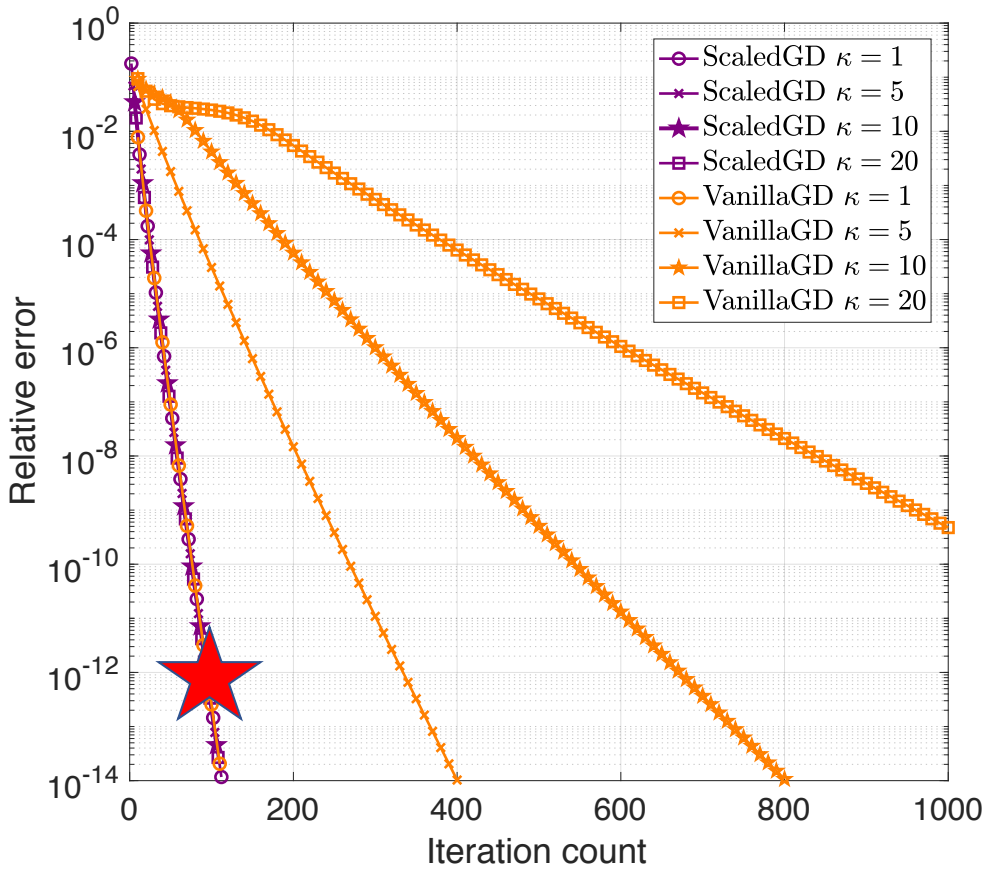
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta \nabla_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

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

for $t = 0, 1, \ldots$

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

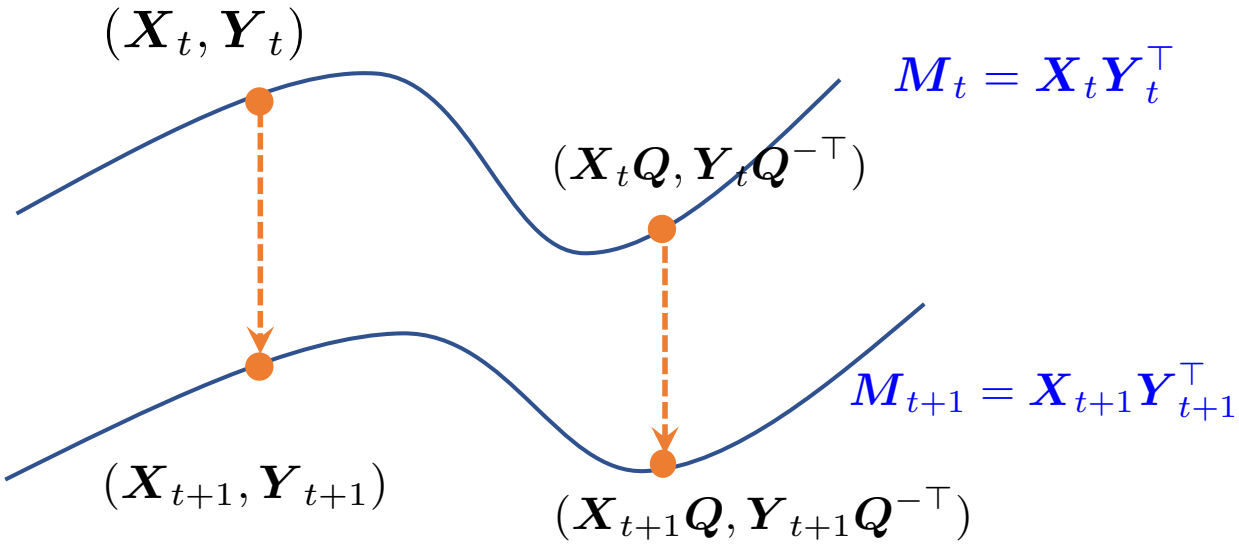
ScaledGD for low-rank matrix completion



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

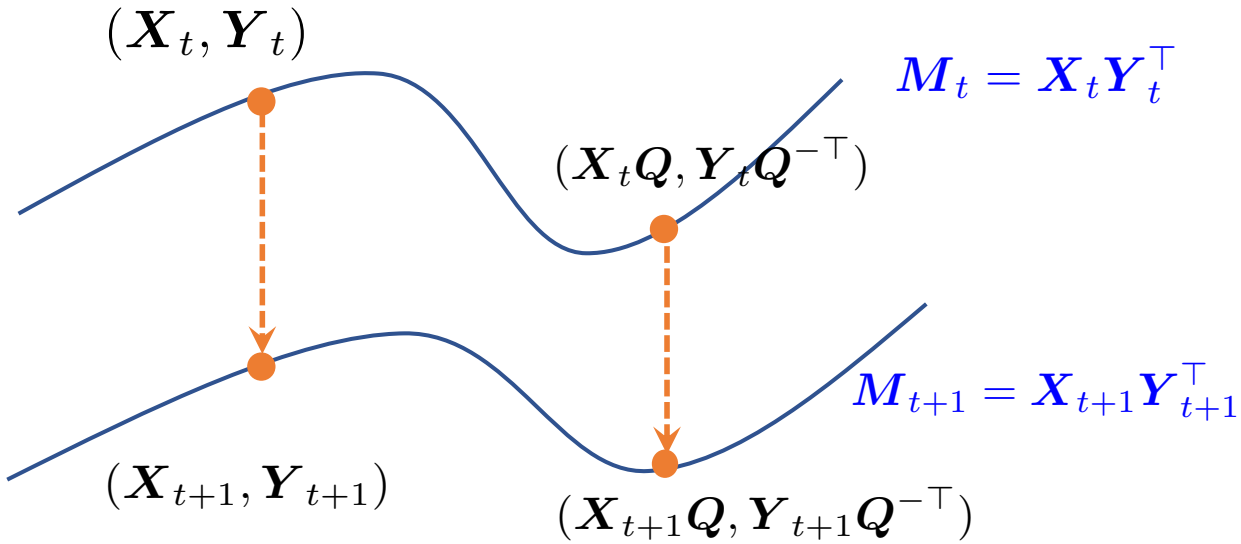
A closer look at ScaledGD

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



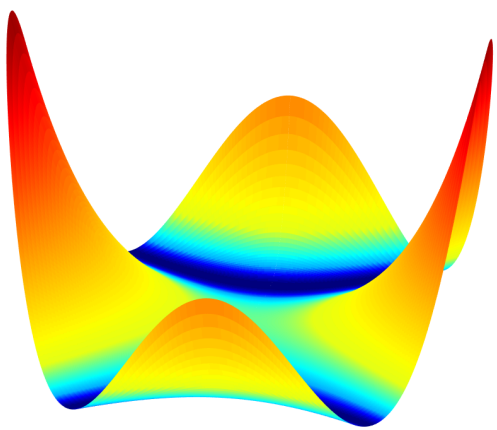
A closer look at ScaledGD

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



New distance metric as Lyapunov function:

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



Theoretical guarantees of ScaledGD

Theorem 2 (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*

$$m \gtrsim nr^2\kappa^2.$$

Theoretical guarantees of ScaledGD

Theorem 2 (Tong, Ma and Chi, 2020)

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

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

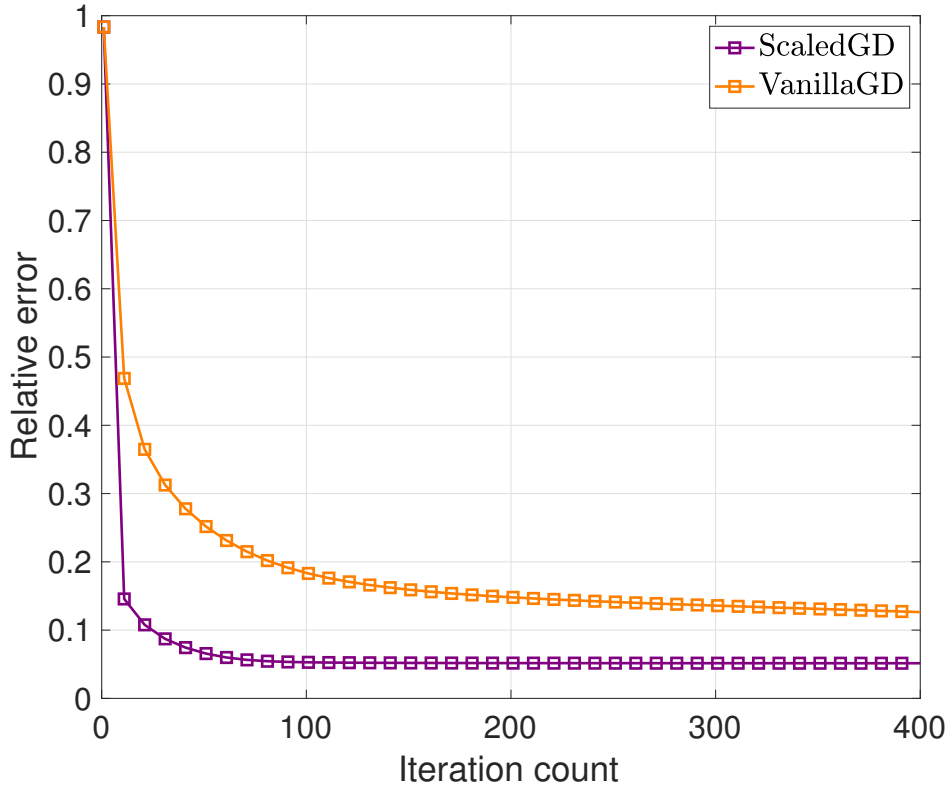
- ***Computational:*** *within* $O\big(\log \frac{1}{\varepsilon}\big)$ *iterations;*
- ***Statistical:*** *the sample complexity satisfies*

$$m \gtrsim nr^2\kappa^2.$$

Compared with Tu et. al.: ScaledGD provably accelerates vanilla GD at the same sample complexity!

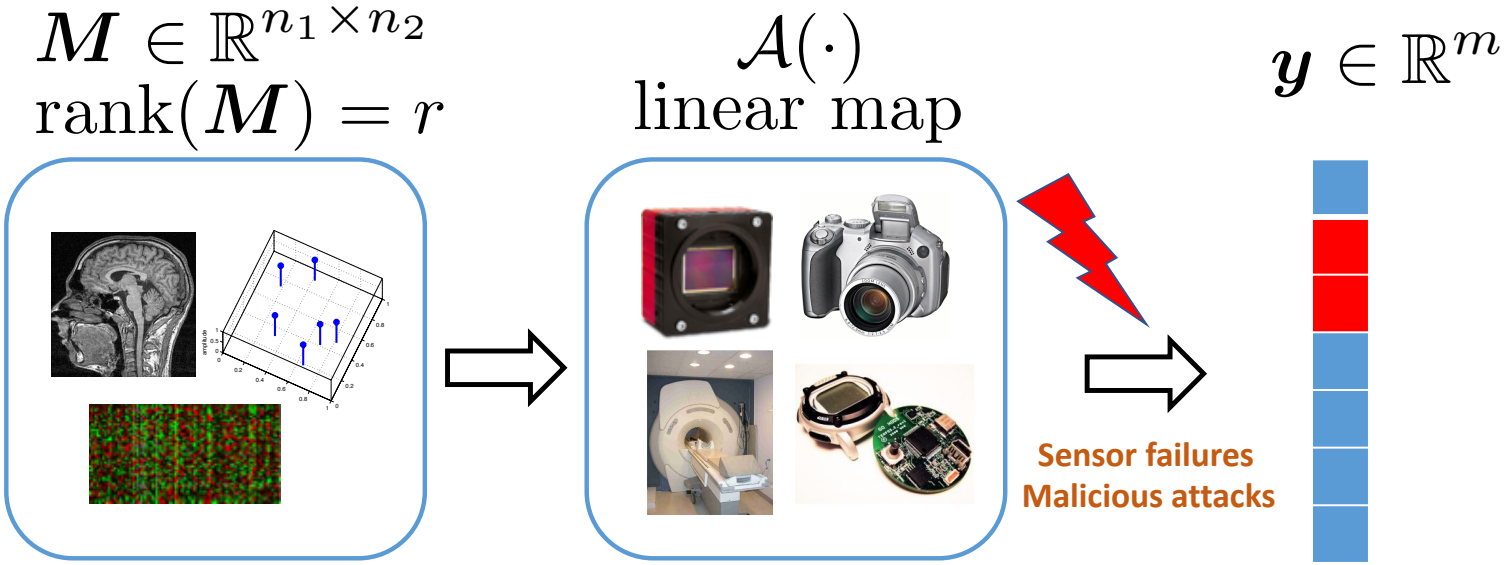
Stability: a numerical result

For the chlorine concentration levels dataset, ScaledGD converges faster than vanilla GD in a small number of iterations.



Robustness to outliers and corruptions?

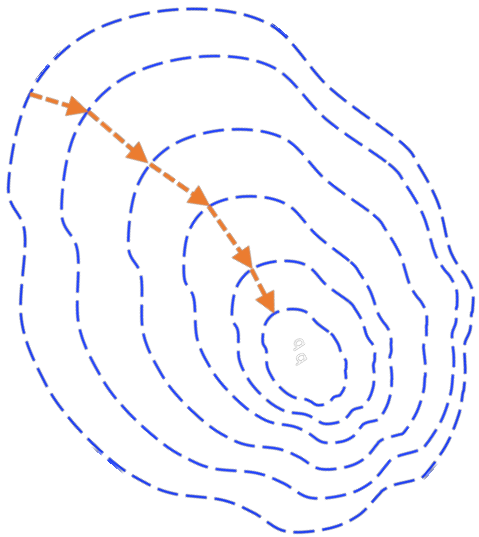
Outlier-corrupted low-rank matrix sensing



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

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

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

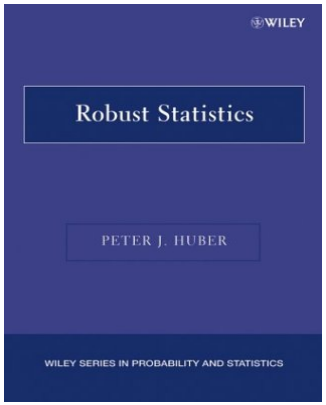
- **Gradient iterations would fail:**

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

for $t = 0, 1, \ldots$

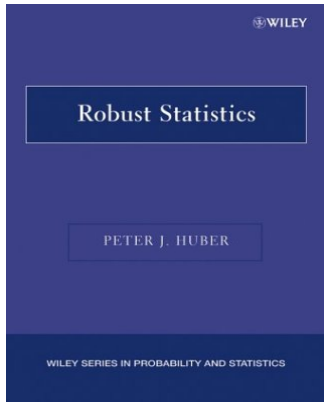
Even a single outlier can fail the algorithm!

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

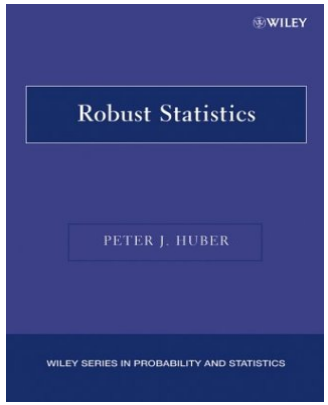


Key idea: "median-truncation" — discard samples *adaptively* based on how large sample gradients / values deviate from median

- **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 3 (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* $O\big(\kappa \log \frac{1}{\varepsilon}\big)$ *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 3 (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* $O\big(\kappa \log \frac{1}{\varepsilon}\big)$ *iterations;*
- ***Statistical:*** *the sample complexity satisfies*

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

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

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

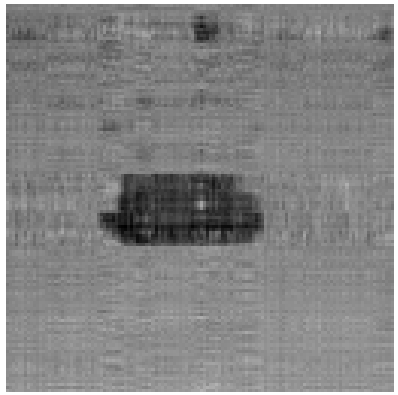
Median-truncated GD adds robustness to GD *obliviously*.

Numerical example

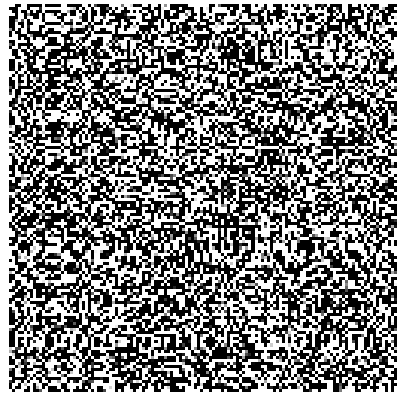
Low-rank matrix sensing:

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

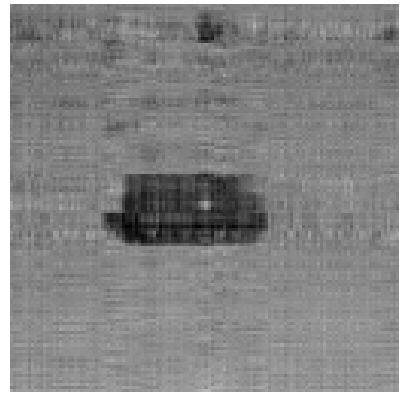
Ground truth



GD
no outliers



GD
1% outliers



median-TGD
1% outliers

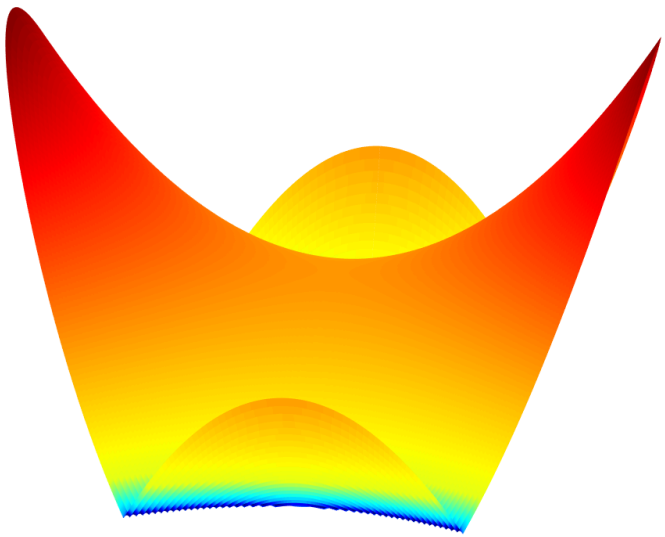
Median-truncated GD achieves similar performance as if performing GD on the clean data.

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:

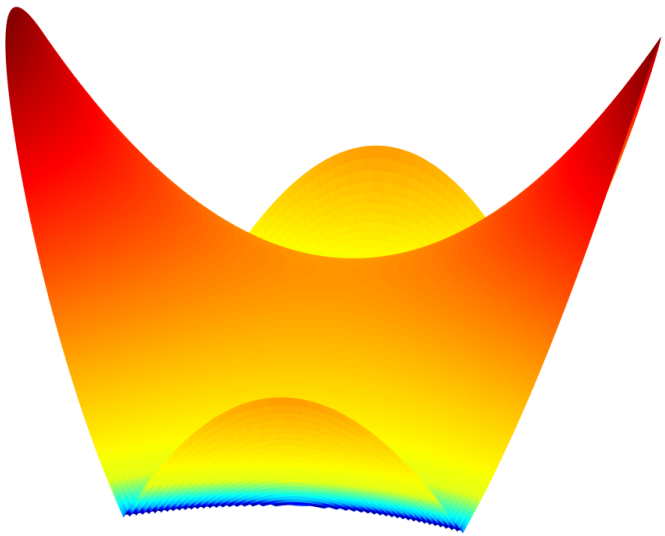
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta_t \, \partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)$$
$$\boldsymbol{Y}_{t+1} = \boldsymbol{Y}_t - \eta_t \, \partial_{\boldsymbol{Y}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t)$$

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:

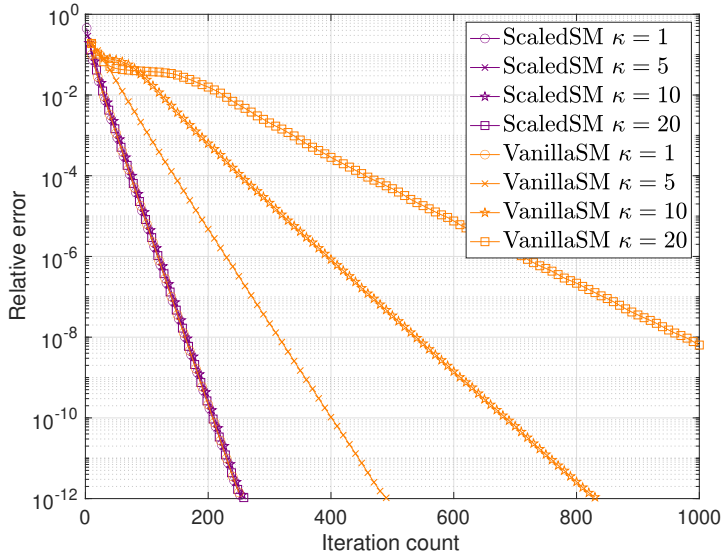
$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t - \eta_t \, \partial_{\boldsymbol{X}} f(\boldsymbol{X}_t, \boldsymbol{Y}_t) \underbrace{(\boldsymbol{Y}_t^\top \boldsymbol{Y}_t)^{-1}}_{\texttt{preconditioner}}$$

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

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

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!

Demixing sparse and low-rank matrices

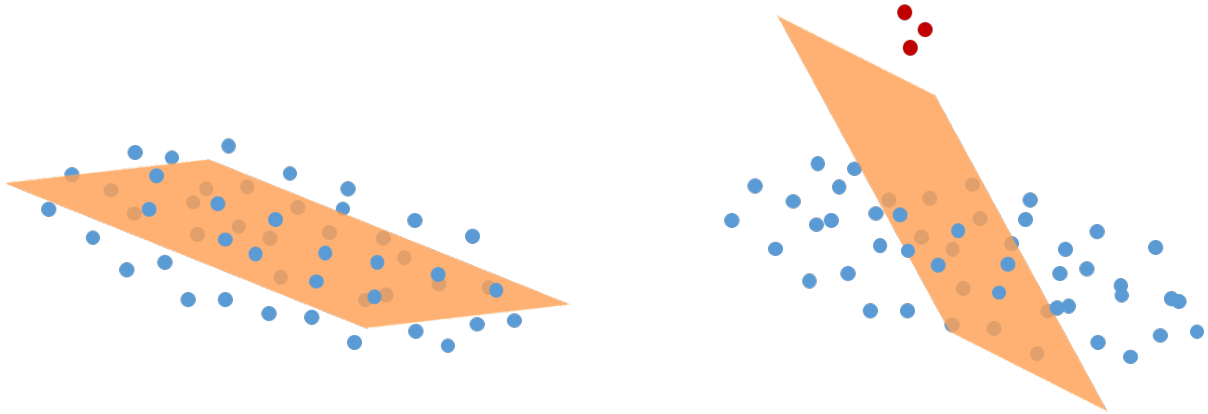
Suppose we are given a matrix

$$\boldsymbol{M} = \underbrace{\boldsymbol{L}}_{\text{low-rank}} + \underbrace{\boldsymbol{S}}_{\text{sparse}} \in \mathbb{R}^{n \times n}$$

Question: can we hope to recover both $\boldsymbol{L}$ and $\boldsymbol{S}$ from $\boldsymbol{M}$?

Applications

- Robust PCA



- Video surveillance: separation of background and foreground

Nonconvex approach

- $\text{rank}(\boldsymbol{L}) \leq r$; if we write the SVD of $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, set

$$\boldsymbol{X}^\star = \boldsymbol{U}_L \boldsymbol{\Sigma}^{1/2}; \quad \boldsymbol{Y}^\star = \boldsymbol{V}\boldsymbol{\Sigma}^{1/2}$$

- non-zero entries of $\boldsymbol{S}$ are "spread out" (no more than s fraction of non-zeros per row/column), but otherwise arbitrary

$$\mathcal{S}_s = \left\{\boldsymbol{S} \in \mathbb{R}^{n\times n} : \quad \|\boldsymbol{S}_{i,:}\|_0 \leq s \cdot n;\ \|\boldsymbol{S}_{:,j}\|_0 \leq s \cdot n\right\}$$

$$\underset{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{S}\in\mathcal{S}_s}{\text{minimize}}\ F(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{S}) := \underbrace{\|\boldsymbol{M} - \boldsymbol{X}\boldsymbol{Y}^\top - \boldsymbol{S}\|_{\mathrm{F}}^2}_{\text{least-squares loss}}$$

where $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{n\times r}$.

Gradient descent and hard thresholding

$$\text{minimize}_{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{S}\in\mathcal{S}_s} \quad F(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{S})$$

- **Spectral initialization**: Set $\boldsymbol{S}^0 = \mathcal{H}_{\gamma s}(\boldsymbol{M})$. Let $\boldsymbol{U}^0\boldsymbol{\Sigma}^0\boldsymbol{V}^{0\top}$ be rank-r SVD of $\boldsymbol{M}^0 := \mathcal{P}_\Omega(\boldsymbol{M}-\boldsymbol{S}^0)$; set $\boldsymbol{X}^0 = \boldsymbol{U}^0\left(\boldsymbol{\Sigma}^0\right)^{1/2}$ and $\boldsymbol{Y}^0 = \boldsymbol{V}^0\left(\boldsymbol{\Sigma}^0\right)^{1/2}$

Gradient descent and hard thresholding

$$\text{minimize}_{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{S}\in\mathcal{S}_s} \quad F(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{S})$$

- **Spectral initialization**: Set $\boldsymbol{S}^0 = \mathcal{H}_{\gamma s}(\boldsymbol{M})$. Let $\boldsymbol{U}^0\boldsymbol{\Sigma}^0\boldsymbol{V}^{0\top}$ be rank-r SVD of $\boldsymbol{M}^0 := \mathcal{P}_\Omega(\boldsymbol{M}-\boldsymbol{S}^0)$; set $\boldsymbol{X}^0 = \boldsymbol{U}^0\left(\boldsymbol{\Sigma}^0\right)^{1/2}$ and $\boldsymbol{Y}^0 = \boldsymbol{V}^0\left(\boldsymbol{\Sigma}^0\right)^{1/2}$
- **for** $t = 0, 1, 2, \cdots$
  - **Hard thresholding**: $\boldsymbol{S}^{t+1} = \mathcal{H}_{\gamma s}(\boldsymbol{M} - \boldsymbol{X}^t\boldsymbol{Y}^{t\top})$
  - **Scaled gradient updates**:

$$\boldsymbol{X}^{t+1} = \boldsymbol{X}^t - \eta\nabla_{\boldsymbol{X}}F\left(\boldsymbol{X}^t,\boldsymbol{Y}^t,\boldsymbol{S}^{t+1}\right)\left(\boldsymbol{Y}_t^\top\boldsymbol{Y}_t\right)^{-1}$$
$$\boldsymbol{Y}^{t+1} = \boldsymbol{Y}^t - \eta\nabla_{\boldsymbol{Y}}F\left(\boldsymbol{X}^t,\boldsymbol{Y}^t,\boldsymbol{S}^{t+1}\right)\left(\boldsymbol{X}_t^\top\boldsymbol{X}_t\right)^{-1}$$

Efficient nonconvex recovery

Theorem 4 (Nonconvex RPCA, Tian, Ma, Chi '20)

Set $\gamma = 2$ *and* $0.1 \le \eta \le 2/3$. *Suppose that*

$$s \lesssim \frac{1}{\mu r^{3/2} \kappa}$$

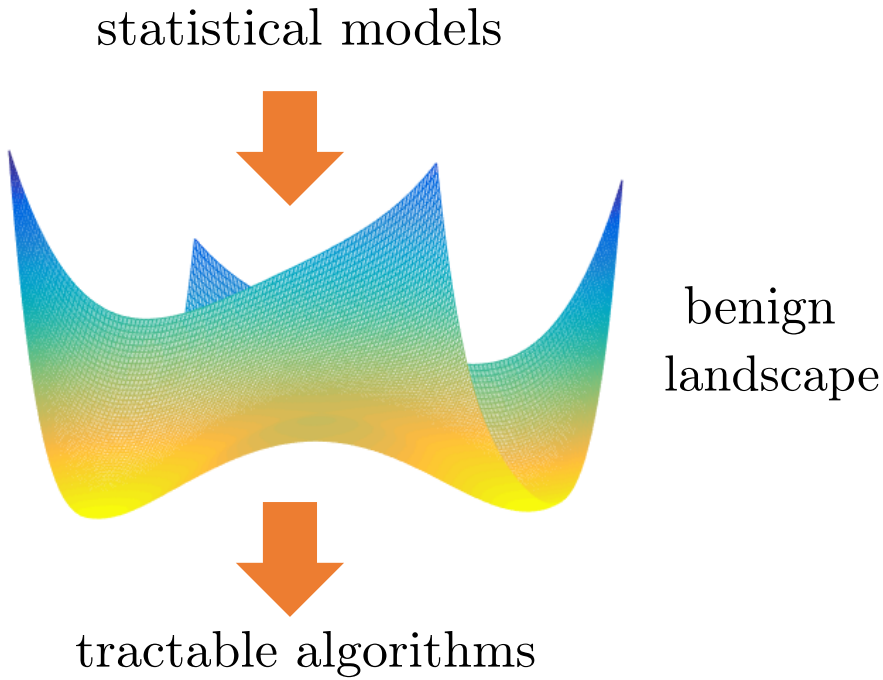
Then GD+HT satisfies

$$\left\| \boldsymbol{X}^t \boldsymbol{Y}^{t\top} - \boldsymbol{L} \right\|_{\mathrm{F}} \lesssim (1 - 0.6\eta)^t \, \sigma_{\min}(\boldsymbol{L})$$

- $O(\log \frac{1}{\varepsilon})$ iterations to reach ε accuracy
- for adversarial outliers, optimal fraction is $s = O(1/\mu r)$; Theorem 4 is suboptimal by a factor of $\kappa\sqrt{r}$
- Improves over GD (Yi et al '16) which requires $s \lesssim \frac{1}{\max\{\mu r^{3/2}\kappa^{3/2}, \mu r \kappa^2\}} \lesssim \frac{1}{\mu r^{3/2}\kappa}$ and $O(\kappa \log \frac{1}{\varepsilon})$ iterations;

Concluding remarks

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
- ...

Open problems

- characterize generic landscape properties that enable fast convergence of gradient methods from random initialization
- relax the stringent assumptions on the statistical models underlying the data
- develop robust and scalable nonconvex methods that can handle distributed data with strong statistical guarantees
- identify new classes of nonconvex problems that admit efficient optimization procedures
- ...

Advertisement: overview and monographs

- "Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview", Y. Chi, Y. M. Lu and Y. Chen.


IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 67, NO. 20, OCTOBER 15, 2019 5239

Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview

Yuejie Chi, Yue M. Lu, and Yuxin Chen

(Overview Article)

- "Spectral Methods for Data Science: A Statistical Perspective", Y. Chen, Y. Chi, J. Fan and C. Ma.

An incomplete list of reference

[1] Dr. Ju Sun's webpage: "`http://sunju.org/research/nonconvex/`".

[2] "*Harnessing Structures in Big Data via Guaranteed Low-Rank Matrix Estimation*," Y. Chen, and Y. Chi, *IEEE Signal Processing Magazine*, 2018.

[3] "*Phase retrieval via Wirtinger flow: Theory and algorithms*," E. Candès, X. Li, M. Soltanolkotabi, *IEEE Transactions on Information Theory*, 2015.

[4] "*Solving random quadratic systems of equations is nearly as easy as solving linear systems*," Y. Chen, E. Candes, Communications on Pure and Applied Mathematics, 2017.

[5] "*Provable non-convex phase retrieval with outliers: Median truncated Wirtinger flow*," H. Zhang, Y. Chi, and Y. Liang, ICML 2016.

[6] "*Implicit Regularization in Nonconvex Statistical Estimation: Gradient Descent Converges Linearly for Phase Retrieval, Matrix Completion and Blind Deconvolution*," C. Ma, K. Wang, Y. Chi and Y. Chen, *Foundations of Computational Mathematics*, 2019.

An incomplete list of reference

[7] "*Gradient Descent with Random Initialization: Fast Global Convergence for Nonconvex Phase Retrieval*," Y. Chen, Y. Chi, J. Fan, C. Ma, *Mathematical Programming*, 2018.

[8] "*Solving systems of random quadratic equations via truncated amplitude flow*," G. Wang, G. Giannakis, and Y. Eldar, *IEEE Transactions on Information Theory*, 2017.

[9] "*Matrix completion from a few entries*," R. Keshavan, A. Montanari, and S. Oh, *IEEE Transactions on Information Theory*, 2010.

[10] "*Guaranteed matrix completion via non-convex factorization*," R. Sun, T. Luo, *IEEE Transactions on Information Theory*, 2016.

[11] "*Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees*," Y. Chen and M. Wainwright, *arXiv preprint arXiv:1509.03025*, 2015.

[12] "*Fast Algorithms for Robust PCA via Gradient Descent*," X. Yi, D. Park, Y. Chen, and C. Caramanis, *NIPS*, 2016.

An incomplete list of reference

[13] "*Matrix completion has no spurious local minimum*," R. Ge, J. Lee, and T. Ma, *NIPS*, 2016.

[14] "*No Spurious Local Minima in Nonconvex Low Rank Problems: A Unified Geometric Analysis*," R. Ge, C. Jin, and Y. Zheng, *ICML*, 2017.

[15] "*Symmetry, Saddle Points, and Global Optimization Landscape of Nonconvex Matrix Factorization*," X. Li *et al.*, *IEEE Transactions on Information Theory*, 2019.

[16] "*Kaczmarz Method for Solving Quadratic Equations*," Y. Chi and Y. M. Lu, *IEEE Signal Processing Letters*, vol. 23, no. 9, pp. 1183-1187, 2016.

[17] "*A Geometric Analysis of Phase Retrieval*," S. Ju, Q. Qu, and J. Wright, to appear, *Foundations of Computational Mathematics*, 2016.

[18] "*Gradient descent converges to minimizers*," J. Lee, M. Simchowitz, M. Jordan, B. Recht, *Conference on Learning Theory*, 2016.

[19] "*Phase retrieval using alternating minimization*," P. Netrapalli, P. Jain, and S. Sanghavi, *NIPS*, 2013.

An incomplete list of reference

[20] "*How to escape saddle points efficiently*," C. Jin, R. Ge, P. Netrapalli, S. Kakade, M. Jordan, arXiv:1703.00887, 2017.

[21] "*Complete dictionary recovery over the sphere*," J. Sun, Q. Qu, J. Wright, *IEEE Transactions on Information Theory*, 2017.

[22] "*A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization*," S. Burer, and R. Monteiro, *Mathematical Programming*, 2003.

[23] "*Memory-efficient Kernel PCA via Partial Matrix Sampling and Nonconvex Optimization: a Model-free Analysis of Local Minima*," J. Chen, X. Li, Journal of Machine Learning Research, 2019.

[24] "*Rapid, robust, and reliable blind deconvolution via nonconvex optimization*," X. Li, S. Ling, T. Strohmer, K. Wei, Applied and computational harmonic analysis, 2019.

[25] "*Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming*," E. Candes, T. Strohmer, V. Voroninski, *Communications on Pure and Applied Mathematics*, 2012.

An incomplete list of reference

[26] "*Exact matrix completion via convex optimization*," E. Candès, B. Recht, *Foundations of Computational mathematics*, 2009.

[27] "*Low-rank solutions of linear matrix equations via Procrustes flow,*" S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, B. Recht, arXiv:1507.03566, 2015.

[28] "*Global optimality of local search for low rank matrix recovery,*" S. Bhojanapalli, B. Neyshabur, and N. Srebro, NIPS, 2016.

[29] "*Optimal rates of convergence for noisy sparse phase retrieval via thresholded Wirtinger flow,*" T. Cai, X. Li, Z. Ma, *The Annals of Statistics*, 2016.

[30] "*The landscape of empirical risk for non-convex losses,*" S. Mei, Y. Bai, and A. Montanari, arXiv:1607.06534, 2016.

[31] "*Non-convex robust PCA,*" P. Netrapalli, U. Niranjan, S. Sanghavi, A. Anandkumar, and P. Jain, NIPS, 2014.

[32] "*Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach,*" D. Park, A. Kyrillidis, C. Caramanis, and S. Sanghavi, arXiv:1609.03240, 2016.

An incomplete list of reference

[33] "*From symmetry to geometry: Tractable nonconvex problems*", Y. Zhang, Q. Qu, J. Wright, arXiv preprint arXiv:2007.06753, 2020.

[34] "*A Nonconvex Approach for Phase Retrieval: Reshaped Wirtinger Flow and Incremental Algorithms,*" H. Zhang, Y. Zhou, Y. Liang and Y. Chi, *Journal of Machine Learning Research*, 2017.

[35] "*Nonconvex Matrix Factorization from Rank-One Measurements,*" Y. Li, C. Ma, Y. Chen and Y. Chi, arXiv:1802.06286, 2018.

[36] "*Manifold Gradient Descent Solves Multi-Channel Sparse Blind Deconvolution Provably and Efficiently*", L. Shi and Y. Chi, arXiv preprint arXiv:1911.11167, 2019.

[37] "*Median-Truncated Gradient Descent: A Robust and Scalable Nonconvex Approach for Signal Estimation*", Y. Chi, Y. Li, H. Zhang, and Y. Liang, Compressed Sensing and Its Applications, Springer, Birkhauser, 2019.

[38] "*Nonconvex Low-Rank Matrix Recovery with Arbitrary Outliers via Median-Truncated Gradient Descent*", Y. Li, Y. Chi, H. Zhang, and Y. Liang, Information and Inference, 2020.

An incomplete list of reference

[39] "*Nonconvex robust low-rank matrix recovery*", Li et al., SIAM Journal on Optimization, 2020.

[40] "*Median-Truncated Nonconvex Approach for Phase Retrieval with Outliers*", H. Zhang, Y. Chi and Y. Liang, IEEE Trans. on Information Theory, 2018.

[41] "*Low-rank matrix recovery with composite optimization: good conditioning and rapid convergence*", V. Charisopoulos et al., arXiv:1904.10020, 2019.

[42] "*Accelerating Ill-Conditioned Low-Rank Matrix Estimation via Scaled Gradient Descent*", T. Tong, C. Ma, and Y. Chi, arXiv:2005.08898, 2020.

[43] "*Low-Rank Matrix Recovery with Scaled Subgradient Methods: Fast and Robust Convergence Without the Condition Number*", T. Tong, C. Ma, and Y. Chi, arXiv:2010.13364, 2020.