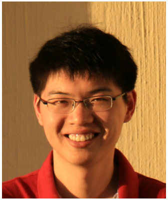
Random Initialization and Implicit Regularization in Nonconvex Statistical Estimation

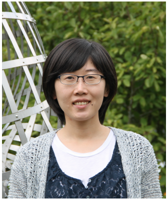
Yuxin Chen

Electrical Engineering, Princeton University

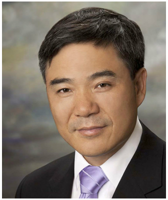


Cong Ma
Princeton ORFE

Kaizheng Wang
Princeton ORFE



Yuejie Chi
CMU ECE

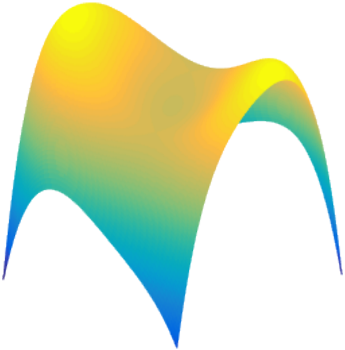


Jianqing Fan
Princeton ORFE

Nonconvex problems are everywhere

Empirical risk minimization is usually nonconvex

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}; \text{data})$$

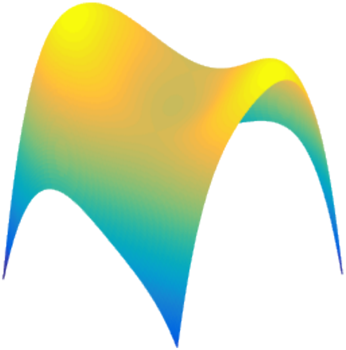


Nonconvex problems are everywhere

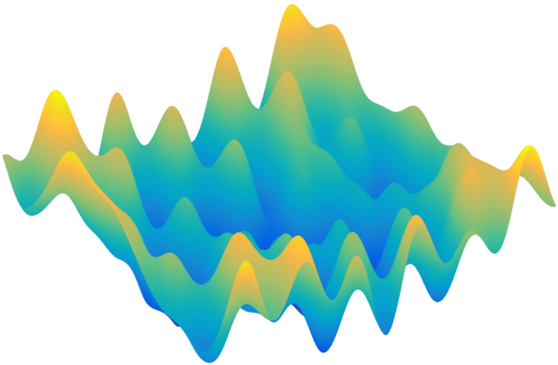
Empirical risk minimization is usually nonconvex

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}; \text{data})$$

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



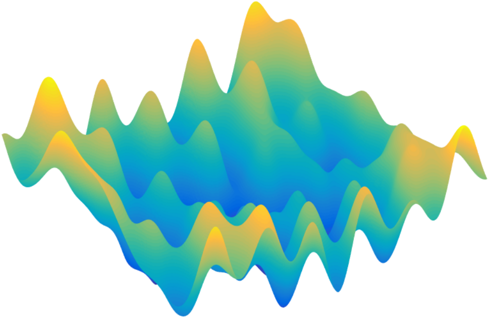
Nonconvex optimization may be super scary

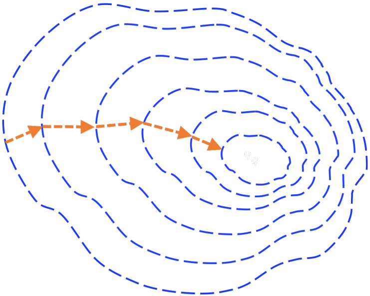


There may be bumps everywhere and exponentially many local optima

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

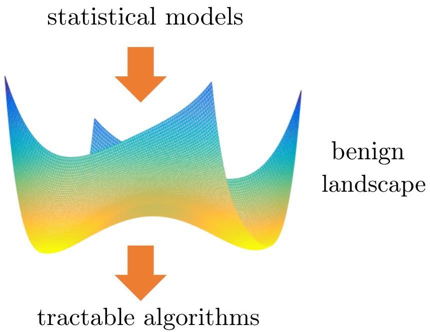
Nonconvex optimization may be super scary





But they are solved on a daily basis via simple algorithms like *(stochastic) gradient descent*

Statistical models come to rescue



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

— *Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview*

Chi, Lu, Chen '18

Example: low-rank matrix recovery

$$\underset{\boldsymbol{U} \in \mathbb{R}^{n \times r}}{\text{minimize}} \quad f(\boldsymbol{U}) := \sum_{i=1}^{m} \left(\langle \boldsymbol{A}_i, \boldsymbol{U}\boldsymbol{U}^\top \rangle - \langle \boldsymbol{A}_i, \boldsymbol{U}^\star \boldsymbol{U}^{\star\top} \rangle \right)^2$$

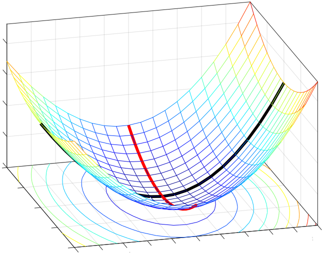
where entries of $\boldsymbol{A}_i$ are i.i.d. Gaussian

Example: low-rank matrix recovery

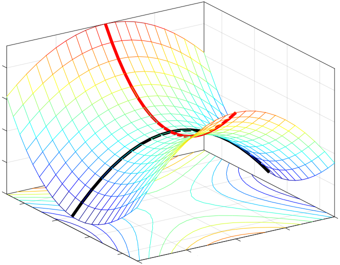
$$\underset{\boldsymbol{U}\in\mathbb{R}^{n\times r}}{\text{minimize}} \quad f(\boldsymbol{U}) := \sum_{i=1}^{m} \left(\langle \boldsymbol{A}_i, \boldsymbol{U}\boldsymbol{U}^\top \rangle - \langle \boldsymbol{A}_i, \boldsymbol{U}^\star \boldsymbol{U}^{\star\top} \rangle \right)^2$$

where entries of $\boldsymbol{A}_i$ are i.i.d. Gaussian

- *no spurious local minima* under large enough sample size (Bhojanapalli et al. '16)



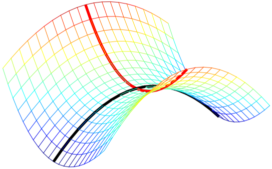
global minimum

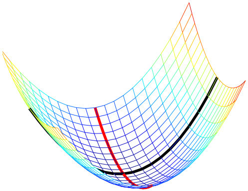


saddle point

Separation of landscape analysis and generic algorithm design

landscape analysis
(statistics)

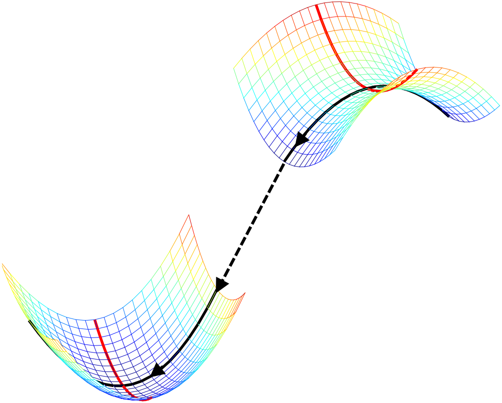




Separation of landscape analysis and generic algorithm design

landscape analysis
(statistics)

generic algorithms
(optimization)



Separation of landscape analysis and generic algorithm design

landscape analysis (statistics)

generic algorithms (optimization)

- 2-layer linear neural network (Baldi, Hornik '89)
- dictionary learning (Sun et al. '15)
- phase retrieval (Sun et al. '16, Davis et al. '17)
- matrix completion (Ge et al. '16, Chen et al. '17)
- matrix sensing (Bhojanapalli et al. '16, Li et al. '16)
- empirical risk mininimization (Mei et al. '16)
- synchronization (Bandeira et al. '16)
- robust PCA (Ge et al. '17)
- inverting deep neural nets (Hand et al. '17)
- 1-hidden-layer neural nets (Ge et al. '17)
- blind deconvolution (Zhang et al. '18, Li et al. '18)
- ...

- cubic regularization (Nesterov, Polyak '06)
- gradient descent (Lee et al. '16)
- trust region method (Sun et al. '16)
- Carmon et al. '16
- perturbed GD (Jin et al. '17)
- perturbed accelerated GD (Jin et al. '17)
- Agarwal et al. '17
- Natasha (Allen-Zhu '17)
- ...

Separation of landscape analysis and generic algorithm design

landscape analysis (statistics)

generic algorithms (optimization)

- 2-layer linear neural network (Baldi, Hornik '89)
- dictionary learning (Sun et al. '15)
- phase retrieval (Sun et al. '16, Davis et al. '17)
- matrix completion (Ge et al. '16, Chen et al. '17)
- matrix sensing (Bhojanapalli et al. '16, Li et al. '16)
- empirical risk mininimization (Mei et al. '16)
- synchronization (Bandeira et al. '16)
- robust PCA (Ge et al. '17)
- inverting deep neural nets (Hand et al. '17)
- 1-hidden-layer neural nets (Ge et al. '17)
- blind deconvolution (Zhang et al. '18, Li et al. '18)
- ...

- cubic regularization (Nesterov, Polyak '06)
- gradient descent (Lee et al. '16)
- trust region method (Sun et al. '16)
- Carmon et al. '16
- perturbed GD (Jin et al. '17)
- perturbed accelerated GD (Jin et al. '17)
- Agarwal et al. '17
- Natasha (Allen-Zhu '17)
- ...

Issue: conservative computational guarantees for specific problems (e.g. solving quadratic systems, matrix completion)

This talk: blending landscape and convergence analysis

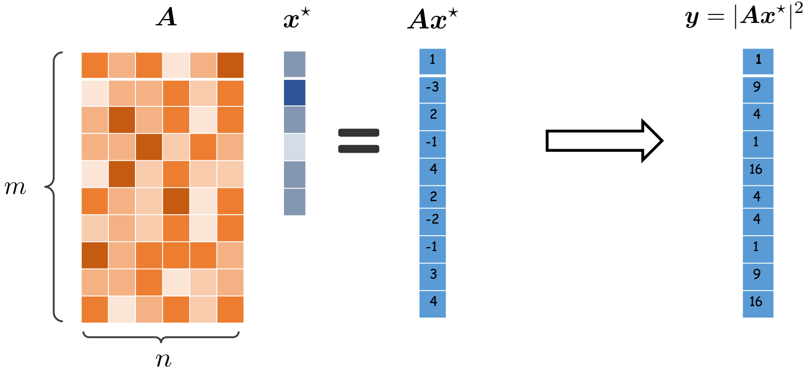
This talk: blending landscape and convergence analysis

⇓

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

A case study: solving random quadratic systems of equations

Solving quadratic systems of equations



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

$$y_k = \left(\boldsymbol{a}_k^\top \boldsymbol{x}^\star\right)^2 + \text{noise}, \qquad k = 1, \ldots, m$$

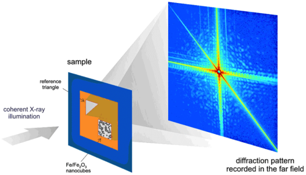
assume w.l.o.g. $\|\boldsymbol{x}^\star\|_2 = 1$

Motivation: phase retrieval

Detectors record intensities of diffracted rays

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

Fig credit: Stanford SLAC



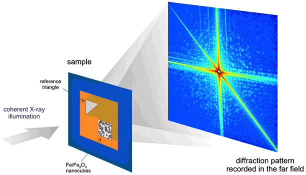
intensity of electrical field: $\left|\widehat{x}(f_1, f_2)\right|^2 = \left|\int x(t_1, t_2) e^{-i2\pi(f_1 t_1 + f_2 t_2)} \mathrm{d}t_1 \mathrm{d}t_2\right|^2$

Motivation: phase retrieval

Detectors record intensities of diffracted rays

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

Fig credit: Stanford SLAC

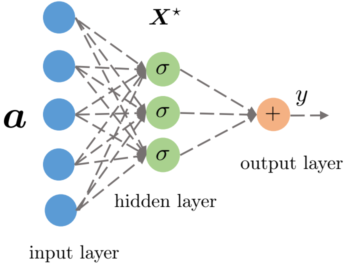


intensity of electrical field: $\left|\widehat{x}(f_1, f_2)\right|^2 = \left|\int x(t_1, t_2) e^{-i2\pi(f_1 t_1 + f_2 t_2)} \mathrm{d}t_1 \mathrm{d}t_2\right|^2$

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*

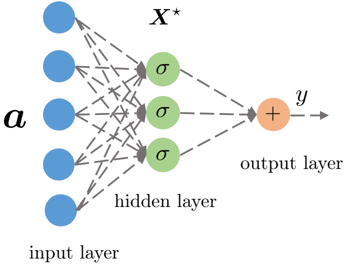


input features: $\boldsymbol{a}$; weights: $\boldsymbol{X}^\star = [\boldsymbol{x}_1^\star, \cdots, \boldsymbol{x}_r^\star]$

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

Motivation: learning neural nets with quadratic activation

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

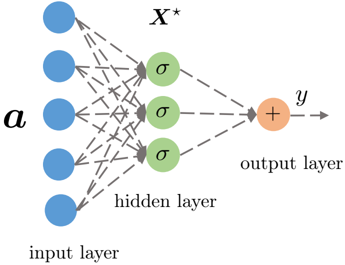


input features: $\boldsymbol{a}$; weights: $\boldsymbol{X}^\star = [\boldsymbol{x}_1^\star, \cdots, \boldsymbol{x}_r^\star]$

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

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 $r = 1$

A natural least squares formulation

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

A natural least squares formulation

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

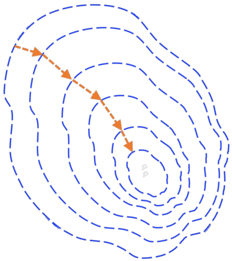
- **issue:** $f(\cdot)$ is highly nonconvex
  $\longrightarrow$ *computationally challenging!*

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)

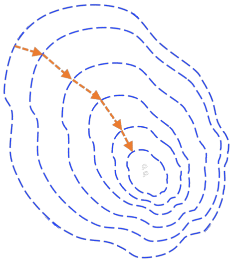
$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4m}\sum_{k=1}^{m}\left[\left(\boldsymbol{a}_k^\top \boldsymbol{x}\right)^2 - y_k\right]^2$$



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

Wirtinger flow (Candès, Li, Soltanolkotabi '14)

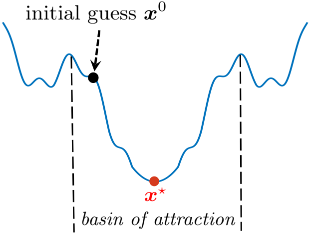
$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) = \frac{1}{4m}\sum_{k=1}^{m}\Big[\big(\boldsymbol{a}_k^\top \boldsymbol{x}\big)^2 - y_k\Big]^2$$



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

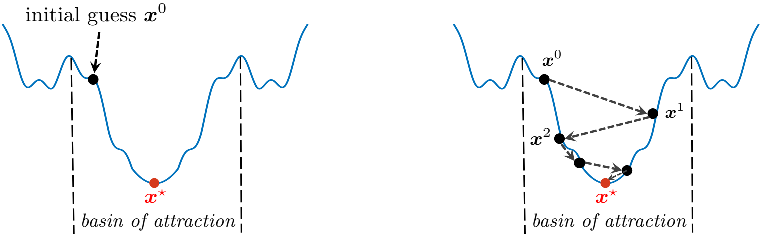
$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta_t \nabla f(\boldsymbol{x}^t), \qquad t = 0, 1, \cdots$$

Rationale of two-stage approach



1. initialize within $\underbrace{\text{local basin sufficiently close to } \boldsymbol{x}^\star}_{\text{(restricted) strongly convex; no saddles / spurious local mins}}$

Rationale of two-stage approach



1. initialize within $\underbrace{\text{local basin sufficiently close to } \boldsymbol{x}^\star}_{\text{(restricted) strongly convex; no saddles / spurious local mins}}$

2. iterative refinement

A highly incomplete list of two-stage methods

phase retrieval:

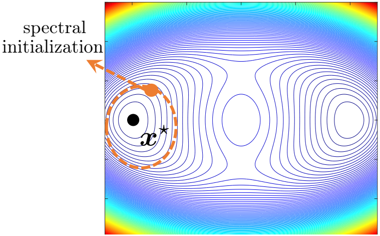
- Netrapalli, Jain, Sanghavi '13
- Candès, Li, Soltanolkotabi '14
- Chen, Candès '15
- Cai, Li, Ma '15
- Wang, Giannakis, Eldar '16
- Zhang, Zhou, Liang, Chi '16
- Kolte, Ozgur '16
- Zhang, Chi, Liang '16
- Soltanolkotabi '17
- Vaswani, Nayer, Eldar '16
- Chi, Lu '16
- Wang, Zhang, Giannakis, Akcakaya, 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
- ...

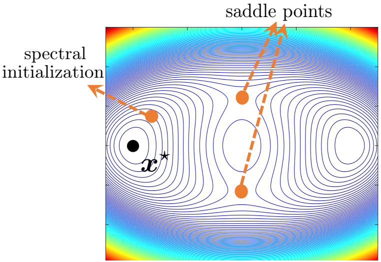
Is carefully-designed initialization necessary for fast convergence?

Initialization



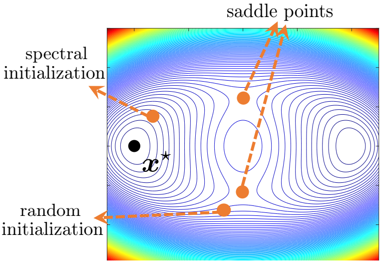
- spectral initialization gets us to (restricted) strongly cvx region

Initialization



- spectral initialization gets us to (restricted) strongly cvx region
- cannot initialize GD anywhere, e.g. might get stuck at saddles

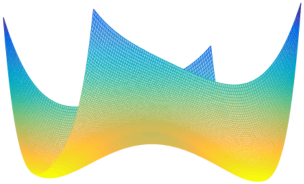
Initialization



- spectral initialization gets us to (restricted) strongly cvx region
- cannot initialize GD anywhere, e.g. might get stuck at saddles

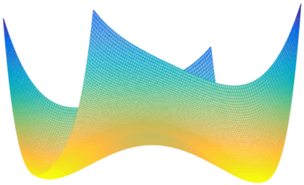
Can we initialize GD randomly, which is simpler and model-agnostic?

What does prior theory say?



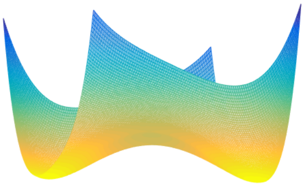
- **landscape:** no spurious local mins (Sun, Qu, Wright '16)

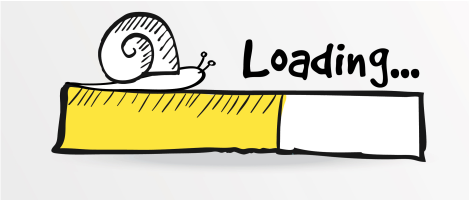
What does prior theory say?



- **landscape:** no spurious local mins (Sun, Qu, Wright '16)
- randomly initialized GD converges almost surely (Lee et al. '16)

What does prior theory say?



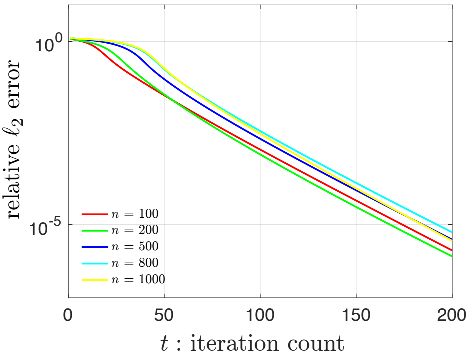


- **landscape:** no spurious local mins (Sun, Qu, Wright '16)
- randomly initialized GD converges almost surely (Lee et al. '16)

"almost surely" might mean "take forever"

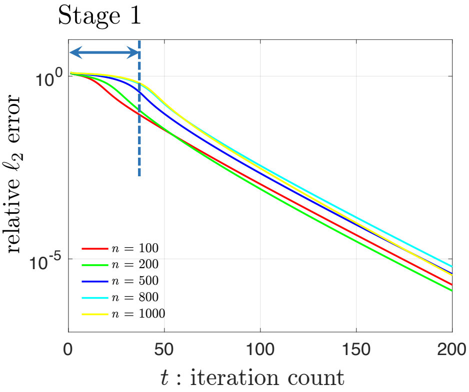
Numerical efficiency of randomly initialized GD

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



Numerical efficiency of randomly initialized GD

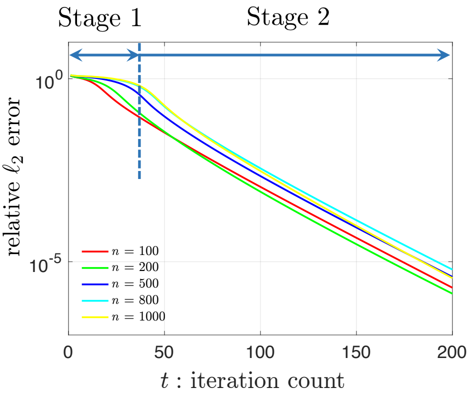
$$\eta = 0.1,\ \boldsymbol{a}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_n),\ m = 10n,\ \boldsymbol{x}^0 \sim \mathcal{N}(\mathbf{0}, n^{-1}\boldsymbol{I}_n)$$



Randomly initialized GD enters local basin within **tens of iterations**

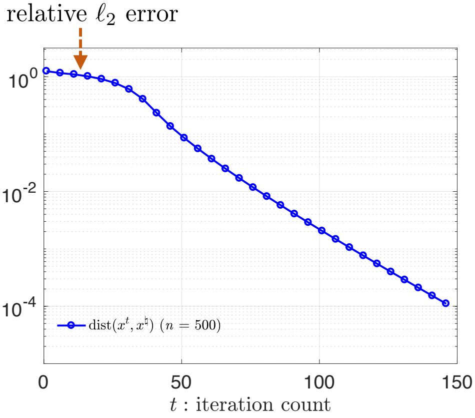
Numerical efficiency of randomly initialized GD

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

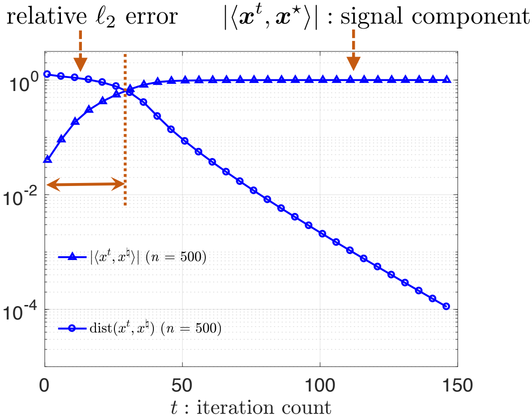


Randomly initialized GD enters local basin within **tens of iterations**

Exponential growth of signal strength in Stage 1



Exponential growth of signal strength in Stage 1



Numerically, a few iterations suffice for entering local region

Our theory: noiseless case

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

Our theory: noiseless case

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

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

Theorem 1 (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*

Our theory: noiseless case

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

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

Theorem 1 (Chen, Chi, Fan, Ma '18)

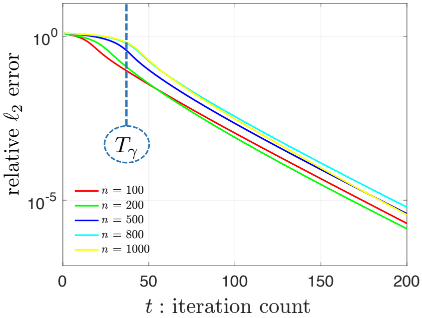
Under i.i.d. Gaussian design, GD with $\boldsymbol{x}^0 \sim \mathcal{N}(\mathbf{0}, n^{-1}\boldsymbol{I}_n)$ achieves

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

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

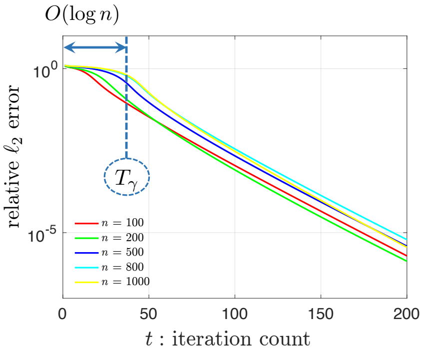
Our theory: noiseless case

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



Our theory: noiseless case

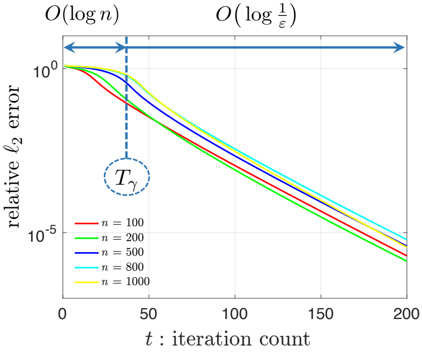
$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \ge T_\gamma \asymp \log n$$



- *Stage 1:* takes $O(\log n)$ iterations to reach $\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma$ (e.g. $\gamma = 0.1$)

Our theory: noiseless case

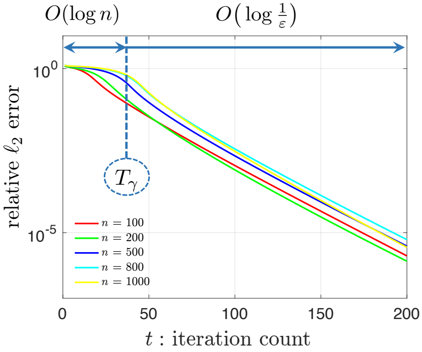
$$\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma(1-\rho)^{t-T_\gamma}\|\boldsymbol{x}^\star\|_2, \quad t \ge T_\gamma \asymp \log n$$



- *Stage 1:* takes $O(\log n)$ iterations to reach $\mathrm{dist}(\boldsymbol{x}^t, \boldsymbol{x}^\star) \le \gamma$ (e.g. $\gamma = 0.1$)
- *Stage 2:* linear (geometric) convergence

Our theory: noiseless case

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

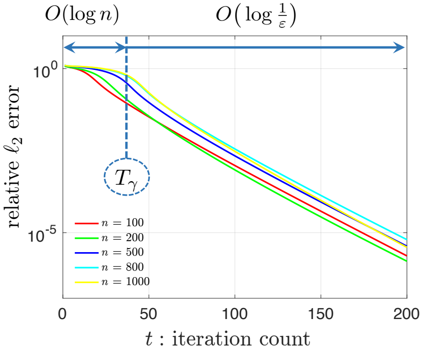


- *near-optimal computational cost*:

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

Our theory: noiseless case

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



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

Stability vis-a-vis noise

$$y_k \quad = \quad |\boldsymbol{a}_k^\top \boldsymbol{x}^\star|^2 + \epsilon_k, \quad \epsilon_k \sim \mathcal{N}(0, \sigma^2) \qquad k = 1, \ldots, m$$

Stability vis-a-vis noise

$$y_k = |\boldsymbol{a}_k^\top \boldsymbol{x}^\star|^2 + \epsilon_k, \quad \epsilon_k \sim \mathcal{N}(0, \sigma^2) \qquad k = 1, \ldots, m$$

- randomly initialized GD converges to **maximum likelihood estimate** in $O\big(\log n + \log \frac{1}{\varepsilon}\big)$ iterations

Stability vis-a-vis noise

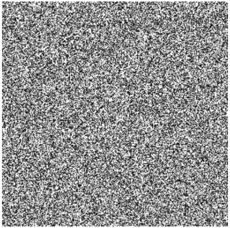
$$y_k \quad = \quad |\boldsymbol{a}_k^\top \boldsymbol{x}^\star|^2 + \epsilon_k, \quad \epsilon_k \sim \mathcal{N}(0, \sigma^2) \qquad k = 1, \ldots, m$$

- randomly initialized GD converges to **maximum likelihood estimate** in $O\big(\log n + \log \frac{1}{\varepsilon}\big)$ iterations
- minimax optimal

Experiments on images

- coded diffraction patterns
- $\boldsymbol{x}^\star \in \mathbb{R}^{256\times 256}$
- $m/n = 12$

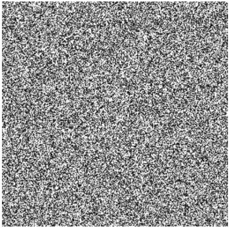
GD with random initialization



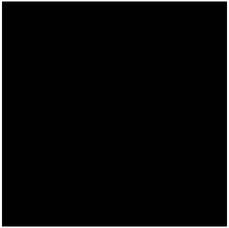
$\boldsymbol{x}^t$
GD iterate

use Adobe to see animation

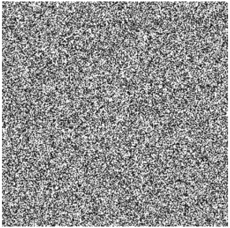
GD with random initialization



$\boldsymbol{x}^t$
GD iterate



$\langle \boldsymbol{x}^t, \boldsymbol{x}^\star \rangle \boldsymbol{x}^\star$
signal component



$\boldsymbol{x}^t - \langle \boldsymbol{x}^t, \boldsymbol{x}^\star \rangle \boldsymbol{x}^\star$
perpendicular component

use Adobe to see animation

Stage 1: random initialization $\rightarrow$ local region

	prior theory based on global landscape	our theory
iteration complexity	almost surely (Lee et al. '16)	$O(\log n)$

What if we have infinite samples?

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

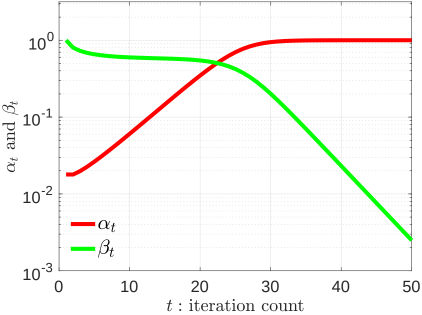
Population level (infinite samples)

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \nabla F(\boldsymbol{x}^t),$$

where

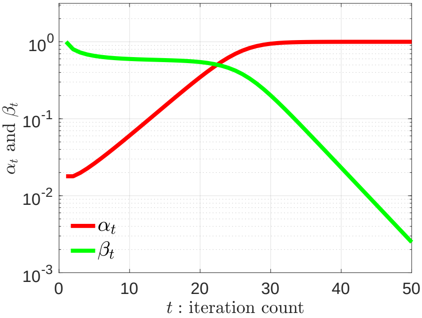
$$\nabla F(\boldsymbol{x}) := \mathbb{E}[\nabla f(\boldsymbol{x})] = (3\|\boldsymbol{x}\|_2^2 - 1)\boldsymbol{x} - 2(\boldsymbol{x}^{\star\top}\boldsymbol{x})\boldsymbol{x}^\star$$

Population-level state evolution



Let $\alpha_t := \underbrace{\left|\langle \boldsymbol{x}^t, \boldsymbol{x}^\star \rangle\right|}_{\text{signal strength}}$ and $\beta_t = \underbrace{\|\boldsymbol{x}^t - \langle \boldsymbol{x}^t, \boldsymbol{x}^\star \rangle \boldsymbol{x}^\star\|_2}_{\text{size of residual component}}$, then

Population-level state evolution



Let $\alpha_t := \underbrace{|\langle \boldsymbol{x}^t, \boldsymbol{x}^\star \rangle|}_{\text{signal strength}}$ and $\beta_t = \underbrace{\|\boldsymbol{x}^t - \langle \boldsymbol{x}^t, \boldsymbol{x}^\star \rangle \boldsymbol{x}^\star\|_2}_{\text{size of residual component}}$, then

$$\alpha_{t+1} = \{1 + 3\eta[1 - (\alpha_t^2 + \beta_t^2)]\}\alpha_t$$
$$\beta_{t+1} = \{1 + \eta[1 - 3(\alpha_t^2 + \beta_t^2)]\}\beta_t$$

2-parameter dynamics

Back to finite-sample analysis

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

Back to finite-sample analysis

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \nabla f(\boldsymbol{x}^t) = \boldsymbol{x}^t - \eta \nabla F(\boldsymbol{x}^t) - \underbrace{\eta\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)}_{\text{residual}}$$

Back to finite-sample analysis

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \nabla f(\boldsymbol{x}^t) = \boldsymbol{x}^t - \eta \nabla F(\boldsymbol{x}^t) - \underbrace{\eta\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)}_{\text{residual}}$$

— take one term in $\boldsymbol{x}^{\star\top}\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)$ as example:

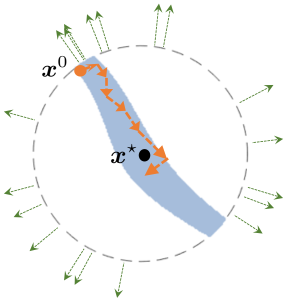
$$\frac{1}{m}\sum_{i=1}^{m} \big(\boldsymbol{a}_i^\top \boldsymbol{x}^t\big)^3 \boldsymbol{a}_i^\top \boldsymbol{x}^\star$$

Back to finite-sample analysis

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \nabla f(\boldsymbol{x}^t) = \boldsymbol{x}^t - \eta \nabla F(\boldsymbol{x}^t) - \underbrace{\eta\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)}_{\text{residual}}$$

— take one term in $\boldsymbol{x}^{\star\top}\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)$ as example:

$$\frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{a}_i^\top \boldsymbol{x}^t\right)^3 \boldsymbol{a}_i^\top \boldsymbol{x}^\star$$



a region with
well-controlled residual

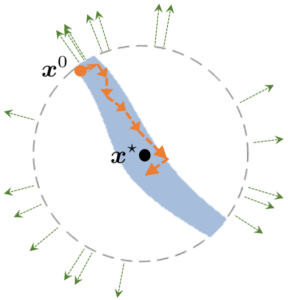
- population-level analysis holds *approximately* if $\boldsymbol{x}^t$ is independent of $\{\boldsymbol{a}_l\}$

Back to finite-sample analysis

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta \nabla f(\boldsymbol{x}^t) = \boldsymbol{x}^t - \eta \nabla F(\boldsymbol{x}^t) - \underbrace{\eta\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)}_{\text{residual}}$$

— take one term in $\boldsymbol{x}^{\star\top}\big(\nabla f(\boldsymbol{x}^t) - \nabla F(\boldsymbol{x}^t)\big)$ as example:

$$\frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{a}_i^\top \boldsymbol{x}^t\right)^3 \boldsymbol{a}_i^\top \boldsymbol{x}^\star$$



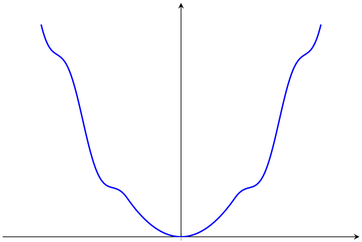
a region with
well-controlled residual

- population-level analysis holds *approximately* if $\boldsymbol{x}^t$ is independent of $\{\boldsymbol{a}_l\}$
- **key analysis ingredient:** show $\boldsymbol{x}^t$ is "nearly-independent" of each $\boldsymbol{a}_l$

Stage 2: local refinement (implicit regularization)

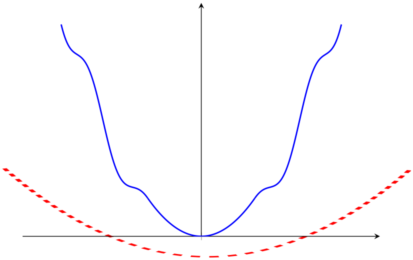
	prior theory	our theory
iteration complexity	$O\left(n \log \frac{1}{\varepsilon}\right)$ (Candès et al. '14)	$O\left(\log \frac{1}{\varepsilon}\right)$

Gradient descent theory revisited



Two standard conditions that enable geometric convergence of GD

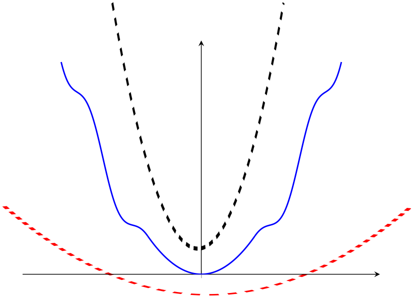
Gradient descent theory revisited



Two standard conditions that enable geometric convergence of GD

- (local) restricted strong convexity

Gradient descent theory revisited



Two standard conditions that enable geometric convergence of GD

- (local) restricted strong convexity
- (local) smoothness

Gradient descent theory revisited

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

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

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 with $\eta = 1/\beta$ obeys

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

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 with $\eta = 1/\beta$ obeys

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

- 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 with $\eta = 1/\beta$ obeys

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

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

What does this optimization theory say about GD?

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 this optimization theory say about GD?

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

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

$$\nabla^2 f(\boldsymbol{x}) \succ 0.5\boldsymbol{I}$$

What does this optimization theory say about GD?

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

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

$$\nabla^2 f(\boldsymbol{x}) \succ 0.5\boldsymbol{I} \quad \underbrace{\text{but ill-conditioned}}_{\text{condition number } \asymp\, n} \text{ (even locally)}$$

What does this optimization theory say about GD?

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

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

$$\nabla^2 f(\boldsymbol{x}) \succ 0.5\boldsymbol{I} \quad \underbrace{\text{but ill-conditioned}}_{\text{condition number } \asymp n} \text{ (even locally)}$$

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

What does this optimization theory say about GD?

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

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

$$\nabla^2 f(\boldsymbol{x}) \succ 0.5\boldsymbol{I} \quad \underbrace{\text{but ill-conditioned}}_{\text{condition number } \asymp\, n} \text{ (even locally)}$$

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

— *optimization theory based on generic landscape conditions implies slow convergence ...*

A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

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

A second look at gradient descent theory

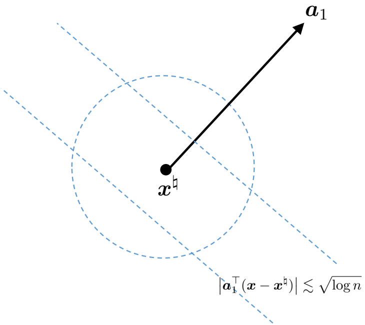
Which local region enjoys both strong convexity and smoothness?

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

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

A second look at gradient descent theory

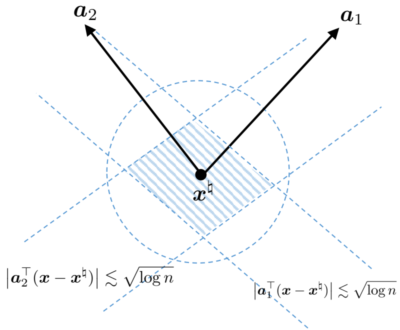
Which local region enjoys both strong convexity and smoothness?



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

A second look at gradient descent theory

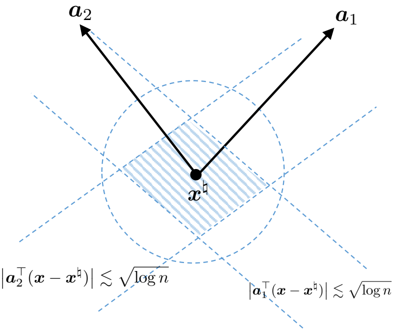
Which local region enjoys both strong convexity and smoothness?



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

A second look at gradient descent theory

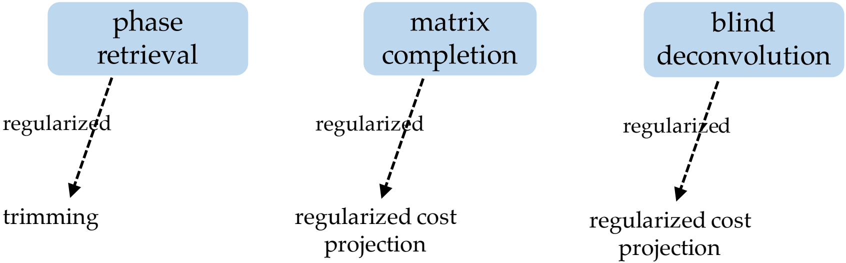
Which local region enjoys both strong convexity and smoothness?



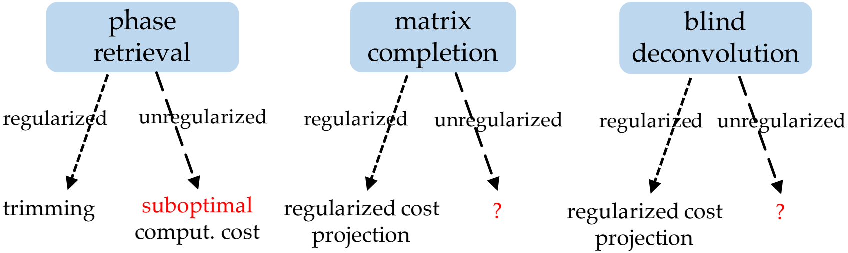
- $\boldsymbol{x}$ is incoherent w.r.t. sampling vectors $\{\boldsymbol{a}_k\}$ (incoherence region)

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

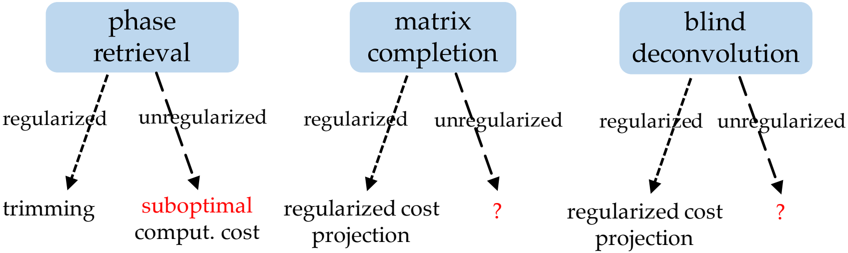
Aside: regularized methods



Aside: regularized vs. unregularized methods

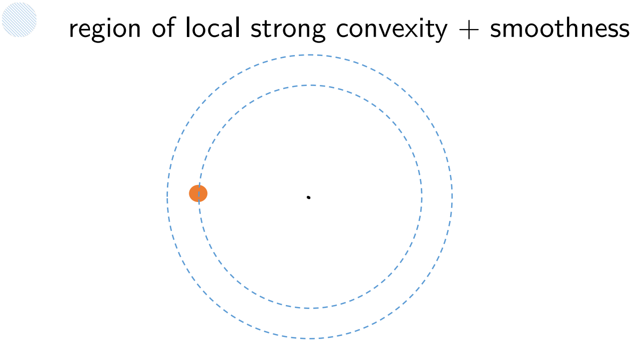


Aside: regularized vs. unregularized methods



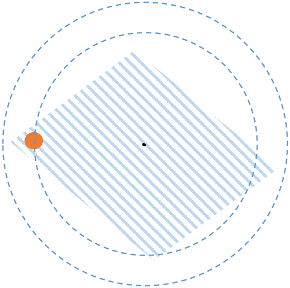
Are unregularized methods suboptimal for nonconvex estimation?

Our findings: GD is implicitly regularized

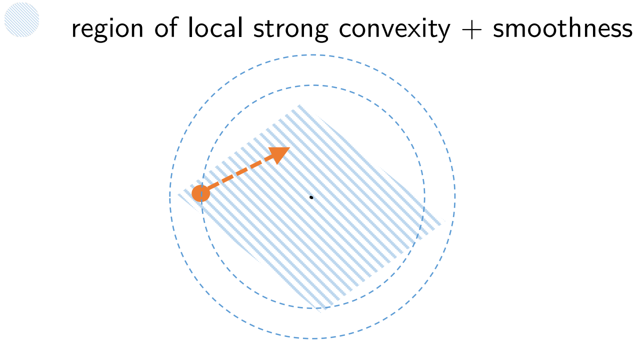


Our findings: GD is implicitly regularized

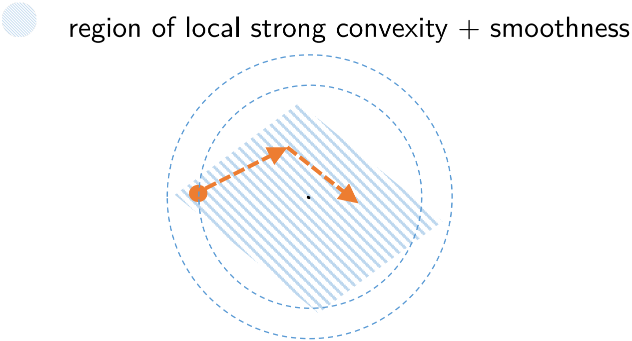
region of local strong convexity + smoothness



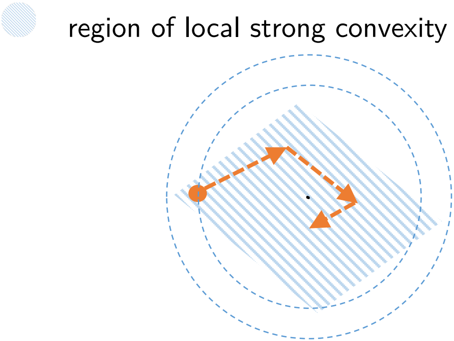
Our findings: GD is implicitly regularized



Our findings: GD is implicitly regularized



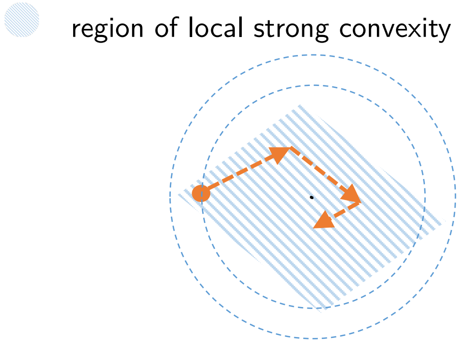
Our findings: GD is implicitly regularized



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

$$\max_l \left|\boldsymbol{a}_l^\top \boldsymbol{x}^t\right| \lesssim \sqrt{\log m}\,\|\boldsymbol{x}^t\|_2, \quad \forall t$$

Our findings: GD is implicitly regularized



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

$$\max_l \left|\boldsymbol{a}_l^\top \boldsymbol{x}^t\right| \lesssim \sqrt{\log m}\,\|\boldsymbol{x}^t\|_2, \quad \forall t$$

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

Key proof idea: leave-one-out analysis

Leave out a small amount of information from data and run GD

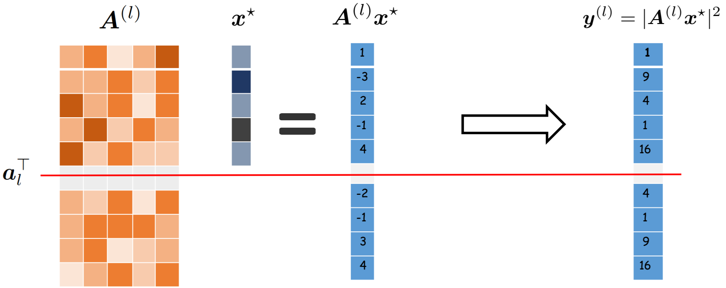
Key proof idea: leave-one-out analysis

Leave out a small amount of information from data and run GD

- Stein '72
- El Karoui, Bean, Bickel, Lim, Yu '13
- El Karoui '15
- Javanmard, Montanari '15
- Zhong, Boumal '17
- Lei, Bickel, El Karoui '17
- Sur, Chen, Candès '17
- Abbe, Fan, Wang, Zhong '17
- Chen, Fan, Ma, Wang '17

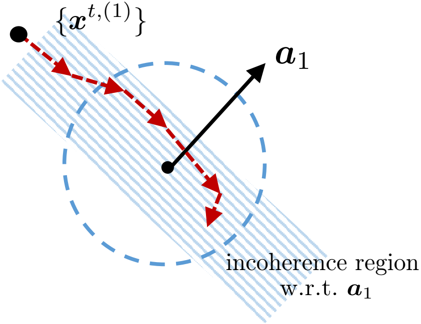
Key proof idea: leave-one-out analysis

Leave out a small amount of information from data and run GD



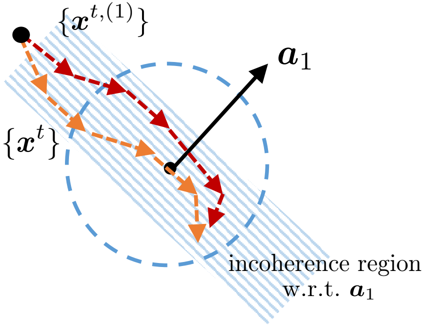
e.g. introduce leave-one-out iterates $\boldsymbol{x}^{t,(l)}$ by running GD without lth sample

Key proof idea: leave-one-out analysis



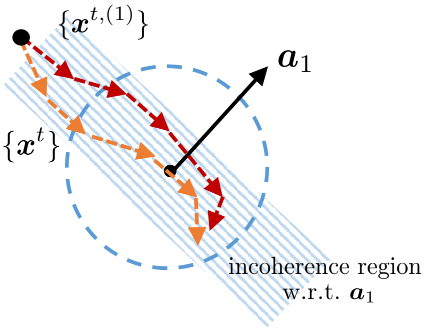
- Leave-one-out iterate $x^{t,(l)}$ is independent of a_l

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 $\boldsymbol{x}^t$

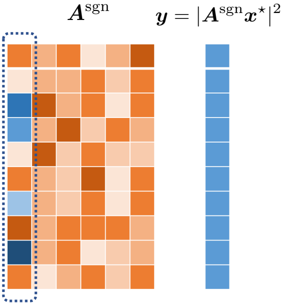
Key proof idea: leave-one-out analysis



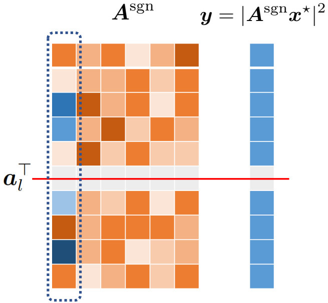
- Leave-one-out iterate $x^{t,(l)}$ is independent of a_l
- Leave-one-out iterate $x^{t,(l)} \approx$ true iterate x^t

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

Key proof ingredient: random-sign sequences



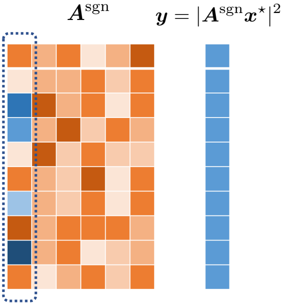
$x^{t,\mathsf{sgn}}$: indep. of sign info of $\{a_{i,1}\}$



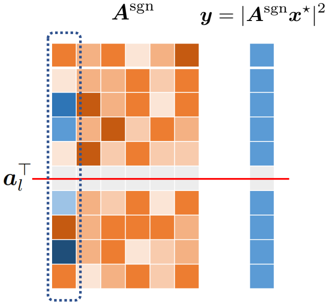
$x^{t,\mathsf{sgn},(l)}$: indep. of both sign info of $\{a_{i,1}\}$ and a_l

- randomly flip signs of $a_i^\top x^\star$ and re-run GD

Key proof ingredient: random-sign sequences



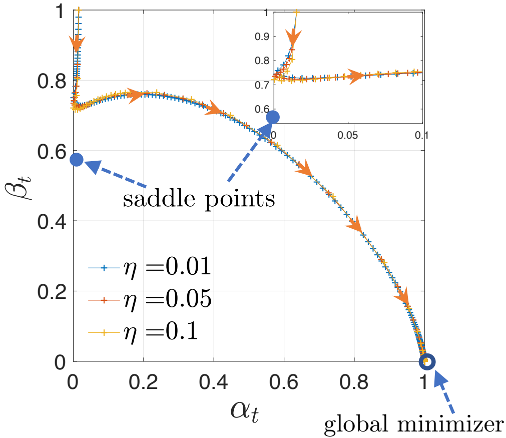
$x^{t,\mathsf{sgn}}$: indep. of sign info of $\{a_{i,1}\}$



$x^{t,\mathsf{sgn},(l)}$: indep. of both sign info of $\{a_{i,1}\}$ and a_l

- randomly flip signs of $a_i^\top x^\star$ and re-run GD
- crucial in controlling $\frac{1}{m}\sum_{i=1}^{m} \left(a_i^\top x^t\right)^3 \underbrace{a_i^\top x^\star}_{|a_i^\top x^\star|\,\mathsf{sgn}(a_i^\top x^\star)}$

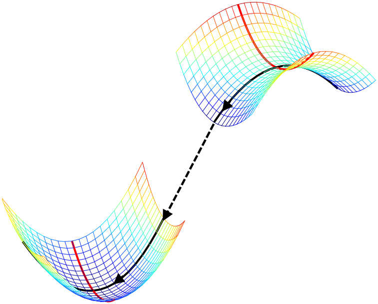
Automatic saddle avoidance



Randomly initialized GD never hits saddle points!

Other saddle-escaping schemes based on generic landscape analysis

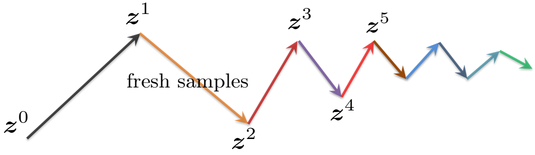
	iteration complexity
trust-region (Sun et al. '16)	$n^7 + \log\log\frac{1}{\varepsilon}$
perturbed GD (Jin et al. '17)	$n^3 + n\log\frac{1}{\varepsilon}$
perturbed accelerated GD (Jin et al. '17)	$n^{2.5} + \sqrt{n}\log\frac{1}{\varepsilon}$
GD (ours) (Chen et al. '18)	$\log n + \log\frac{1}{\varepsilon}$



Generic optimization theory yields highly suboptimal convergence guarantees

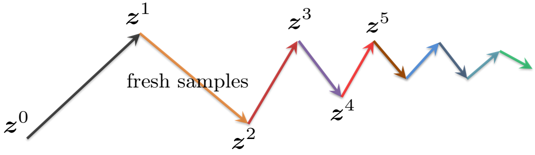
No need of sample splitting

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

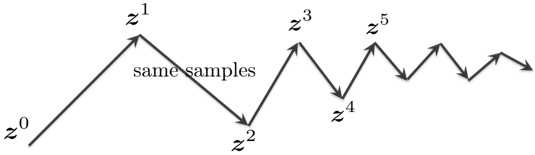


No need of sample splitting

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



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



Concluding remarks

Even simplest nonconvex methods
are remarkably efficient under suitable statistical models

smart initialization	extra regularization	sample splitting	saddle escaping
			

1. "Gradient Descent with Random Initialization: ...", Y. Chen, Y. Chi, J. Fan, C. Ma, *Mathematical Programming*, vol. 176, no. 1-2, pp. 5-37, July 2019

2. "Implicit regularization in nonconvex statistical estimation: ...", C. Ma, K. Wang, Y. Chi, Y. Chen, accepted to *Foundations of Computational Mathematics*, 2019

3. "Nonconvex optimization meets low-rank matrix factorization: An overview", Y. Chi, Y. Lu, Y. Chen, accepted to *IEEE Trans. Signal Processing*, 2019