Random Initialization and Implicit Regularization in Nonconvex Statistical Estimation

Yuxin Chen

Electrical Engineering, Princeton University
Nonconvex problems are everywhere

Empirical risk minimization is usually nonconvex

$$\minimize_x f(x; y)$$
Nonconvex problems are everywhere

Empirical risk minimization is usually nonconvex

$$\min_{x} \ f(x; y)$$

- 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)
Nonconvex optimization may be super scary

But they are solved on a daily basis via simple algorithms like (stochastic) gradient descent
When data are generated by certain statistical models, problems are often much nicer than worst-case instances.
Example: solving random quadratic systems

minimize $\mathbf{x} \in \mathbb{R}^n$  \[ \sum_{k=1}^{m} \left[ (a_k^T \mathbf{x})^2 - (a_k^T \mathbf{x}_0)^2 \right]^2 \]
Example: solving random quadratic systems

\[
\minimize_{\mathbf{x} \in \mathbb{R}^n} \sum_{k=1}^{m} \left[ (a_k^\top \mathbf{x})^2 - (a_k^\top \mathbf{x}^h)^2 \right]^2
\]

where \( a_k \) i.i.d. \( \sim \mathcal{N}(0, I_n) \)
Example: solving random quadratic systems

\[
\text{minimize}_{x \in \mathbb{R}^n} \quad \sum_{k=1}^{m} \left[ (a_k^T x)^2 - (a_k^T x^*)^2 \right]^2
\]

where \( a_k \sim \mathcal{N}(0, I_n) \)

- **no spurious local minima** for large enough \( m \) (Sun et al. ’16)
Separation of landscape analysis and generic algorithm design

landscape analysis (statistics)

• dictionary learning (Sun et al. '15)
• phase retrieval (Sun et al. '16)
• matrix completion (Ge et al. '16, Chen et al. '17)
• matrix sensing (Bhojanapalli et al. '16)
• empirical risk minimization (Mei et al. '16)
• synchronization (Bandeira et al. '16)
• inverting deep neural nets (Hand et al. '17)
• ...

• gradient descent (Lee et al. '16)
• trust region method (Sun et al. '16)
• perturbed GD (Jin et al. '17)
• perturbed accelerated GD (Jin et al. '17)
• cubic regularization (Agarwal et al. '17)
• Natasha (Allen-Zhu '17)
• cubic-regularized Newton (Carmon et al. '16)
• ...

Issue: conservative computational guarantees for specific problems (e.g. solving quadratic systems, matrix completion)
Separation of landscape analysis and generic algorithm design

landscape analysis (statistics)

generic algorithms (optimization)

• dictionary learning (Sun et al. '15)
• phase retrieval (Sun et al. '16)
• matrix completion (Ge et al. '16, Chen et al. '17)
• matrix sensing (Bhojanapalli et al. '16)
• empirical risk minimization (Mei et al. '16)
• synchronization (Bandeira et al. '16)
• inverting deep neural nets (Hand et al. '17)

• gradient descent (Lee et al. '16)
• trust region method (Sun et al. '16)
• perturbed GD (Jin et al. '17)
• perturbed accelerated GD (Jin et al. '17)
• cubic regularization (Agarwal et al. '17)
• Natasha (Allen-Zhu '17)
• cubic-regularized Newton (Carmon et al. '16)

Issue: conservative computational guarantees for specific problems (e.g. solving quadratic systems, matrix completion)
Separation of landscape analysis and generic algorithm design

- dictionary learning (Sun et al. '15)
- phase retrieval (Sun et al. '16)
- matrix completion (Ge et al. '16, Chen et al. '17)
- matrix sensing (Bhojanapalli et al. '16)
- empirical risk minimization (Mei et al. '16)
- synchronization (Bandeira et al. '16)
- inverting deep neural nets (Hand et al. '17)
- ...

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

Issue: conservative computational guarantees for specific problems (e.g. solving quadratic systems, matrix completion)
Separation of landscape analysis and generic algorithm design

landscape analysis (statistics)

- dictionary learning (Sun et al. '15)
- phase retrieval (Sun et al. '16)
- matrix completion (Ge et al. '16, Chen et al. '17)
- matrix sensing (Bhojanapalli et al. '16)
- empirical risk minimization (Mei et al. '16)
- synchronization (Bandeira et al. '16)
- inverting deep neural nets (Hand et al. '17)
- ...

generic algorithms (optimization)

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

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
Outline

- Phase retrieval / solving random quadratic systems of equations
- Matrix completion
Outline

• Phase retrieval / solving random quadratic systems of equations

• Matrix completion
Missing phase problem

Detectors record **intensities** of diffracted rays

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

**Fig credit:** Stanford SLAC

Intensity of electrical field:

$$|\hat{x}(f_1, f_2)|^2 = \left| \int x(t_1, t_2)e^{-i2\pi(f_1 t_1 + f_2 t_2)} dt_1 dt_2 \right|^2$$
Missing phase problem

Detectors record intensities of diffracted rays

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

*Fig credit: Stanford SLAC*

Intensity of electrical field:

$$|\hat{x}(f_1, f_2)|^2 = \left| \int x(t_1, t_2)e^{-i2\pi(f_1t_1 + f_2t_2)} dt_1 dt_2 \right|^2$$

**Phase retrieval:** recover signal $x(t_1, t_2)$ from intensity $|\hat{x}(f_1, f_2)|^2$
Solving quadratic systems of equations

Recover \( x^\dagger \in \mathbb{R}^n \) from \( m \) random quadratic measurements

\[
y_k = (a_k^\top x^\dagger)^2, \quad k = 1, \ldots, m
\]

assume w.l.o.g. \( \|x^\dagger\|_2 = 1 \)
A natural least squares formulation

\[
\begin{align*}
\text{minimize}_{x \in \mathbb{R}^n} \quad & f(x) = \frac{1}{4m} \sum_{k=1}^{m} \left[ (a_k^T x)^2 - y_k \right]^2 \\
\end{align*}
\]
A natural least squares formulation

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

• **pros:** often exact as long as sample size is sufficiently large
A natural least squares formulation

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

- **pros:** often exact as long as sample size is sufficiently large
- **cons:** \( f(\cdot) \) is highly nonconvex
  \[\rightarrow \text{computationally challenging!} \]
Wirtinger flow (Candès, Li, Soltanolkotabi '14)

\[ \text{minimize}_x \quad f(x) = \frac{1}{4m} \sum_{k=1}^{m} \left[ (a_k^\top x)^2 - y_k \right]^2 \]
Wirtinger flow (Candès, Li, Soltanolkotabi '14)

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

- **spectral initialization**: \( x^0 \leftarrow \text{leading eigenvector of certain data matrix} \)
Wirtinger flow (Candès, Li, Soltanolkotabi ’14)

\[ \text{minimize}_x \quad f(x) = \frac{1}{4m} \sum_{k=1}^{m} \left[ (a_k^\top x)^2 - y_k \right]^2 \]

- **spectral initialization:** \( x^0 \leftarrow \) leading eigenvector of certain data matrix

- **gradient descent:**
  \[ x^{t+1} = x^t - \eta_t \nabla f(x^t), \quad t = 0, 1, \cdots \]
Rationale of two-stage approach

1. initialize within local basin sufficiently close to $x^\dagger$

(restricted) strongly convex; no saddles / local mins
Rationale of two-stage approach

1. initialize within basin of attraction sufficiently close to \( \mathbf{x}^\dagger \)
   \( \text{(restricted) strongly convex; no saddles / local mins} \)

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?
• spectral initialization gets us to (restricted) strongly cvx region
• 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 et al.’16)
What does prior theory say?

- **landscape**: no spurious local mins (Sun et al. ’16)
- randomly initialized GD converges **almost surely** (Lee et al. ’16)
What does prior theory say?

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

“almost surely” might mean “takes forever”
Numerical efficiency of randomly initialized GD

\[ \eta = 0.1, \ a_i \sim \mathcal{N}(0, I_n), \ m = 10n, \ x^0 \sim \mathcal{N}(0, n^{-1} I_n) \]
Numerical efficiency of randomly initialized GD

\[ \eta = 0.1, \ a_i \sim \mathcal{N}(0, I_n), \ m = 10n, \ x^0 \sim \mathcal{N}(0, n^{-1}I_n) \]

Randomly initialized GD enters local basin within a few iterations
Numerical efficiency of randomly initialized GD

\[ \eta = 0.1, \ a_i \sim \mathcal{N}(0, I_n), \ m = 10n, \ x^0 \sim \mathcal{N}(0, n^{-1} I_n) \]

Randomly initialized GD enters local basin within a few iterations
Exponential growth of signal strength in Stage 1

Relative $\ell_2$ error

$\text{dist}(x^t, x^\star)$ ($n = 500$)

t : iteration count
Exponential growth of signal strength in Stage 1

Relative $\ell_2$ error $|\langle x^t, x^i \rangle|$ : signal component

Numerically, a few iterations suffice for entering local region.
Exponential growth of signal strength in Stage 1

Numerically, a few iterations suffice for entering local region
Our theory

These numerical findings can be formalized when $a_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n)$:
Our theory

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

\[
\text{dist}(\mathbf{x}^t, \mathbf{x}^\sharp) := \min\{\|\mathbf{x}^t \pm \mathbf{x}^\sharp\|_2\}
\]

**Theorem 1 (Chen, Chi, Fan, Ma ’18)**

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

These numerical findings can be formalized when $a_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n)$:

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

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

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

$$\text{dist}(x^t, x^\|) \leq \gamma (1 - \rho)^{t - T\gamma} \|x^\|$ \|_2, \quad t \geq T\gamma$$

for $T\gamma \lesssim \log n$ and some constants $\gamma, \rho > 0$, provided that step size $\eta \asymp 1$ and sample size $m \gtrsim n \text{ polylog } m$.
Our theory

\[
\text{dist}(\mathbf{x}^t, \mathbf{x}^\dagger) \leq \gamma (1 - \rho)^{t - T\gamma} \|\mathbf{x}^\dagger\|_2, \quad t \geq T\gamma \asymp \log n
\]
Our theory

\[ \text{dist}(\mathbf{x}^t, \mathbf{x}^h) \leq \gamma (1 - \rho)^{t - T_\gamma} \| \mathbf{x}^h \|_2, \quad t \geq T_\gamma \asymp \log n \]

- **Stage 1**: takes \( O(\log n) \) iterations to reach \( \text{dist}(\mathbf{x}^t, \mathbf{x}^h) \leq \gamma \)
Our theory

\[
\text{dist}(x^t, x^\dagger) \leq \gamma (1 - \rho)^{t - T_\gamma} \|x^\dagger\|_2, \quad t \geq T_\gamma \approx \log n
\]

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

- **Stage 2**: linear convergence

![Graph showing convergence rate](image-url)
Our theory

\[ \text{dist}(\mathbf{x}^t, \mathbf{x}^\parallel) \leq \gamma (1 - \rho)^{t - T_\gamma} \| \mathbf{x}^\parallel \|_2, \quad t \geq T_\gamma \approx \log n \]

- **near-optimal computational cost:**
  \[ O(\log n + \log \frac{1}{\varepsilon}) \] iterations to yield \( \varepsilon \) accuracy
Our theory

\[
\text{dist}(\mathbf{x}^t, \mathbf{x}^\dagger) \leq \gamma (1 - \rho)^{t - T_{\gamma}} \|\mathbf{x}^\dagger\|_2, \quad t \geq T_{\gamma} \gtrsim \log n
\]

- **near-optimal computational cost:**
  \[
  O(\log n + \log \frac{1}{\varepsilon}) \] iterations to yield \(\varepsilon\) accuracy

- **near-optimal sample size:** \(m \gtrsim n \text{poly } \log m\)
Experiments on images

- coded diffraction patterns
- $\mathbf{x}^h \in \mathbb{R}^{256 \times 256}$
- $m/n = 12$
GD with random initialization

$\mathbf{x}^t$
GD iterate

use Adobe to see animation
GD with random initialization

$x^t$
GD iterate

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

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

*use Adobe to see animation*
Stage 1: random initialization $\rightarrow$ local region

| iteration complexity | prior theory based on global landscape (Lee et al. '16) | our theory $O(\log n)$ |
What if we have infinite samples?

Gaussian designs: \( a_k \overset{i.i.d.}{\sim} \mathcal{N}(0, I_n), \quad 1 \leq k \leq m \)

Population level (infinite samples)

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

where

\[
\nabla F(x) := \mathbb{E}[\nabla f(x)] = (3\|x\|_2^2 - 1)x - 2(x^\top x)x^\parallel
\]
Let $\alpha_t := \left| \langle \mathbf{x}^t, \mathbf{x}^{\dagger} \rangle \right|$ and $\beta_t = \| \mathbf{x}^t - \langle \mathbf{x}^t, \mathbf{x}^{\dagger} \rangle \mathbf{x}^{\dagger} \|_2$, then
Let $\alpha_t := \left| \langle \mathbf{x}^t, \mathbf{x}^\dagger \rangle \right|$ and $\beta_t = \| \mathbf{x}^t - \langle \mathbf{x}^t, \mathbf{x}^\dagger \rangle \mathbf{x}^\dagger \|_2$, 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

\[ x^{t+1} = x^t - \eta \nabla f(x^t) \]
Back to finite-sample analysis

\[ x^{t+1} = x^t - \eta \nabla f(x^t) = x^t - \eta \nabla F(x^t) - \eta (\nabla f(x^t) - \nabla F(x^t)) \]

residual
Back to finite-sample analysis

\[ x^{t+1} = x^t - \eta \nabla f(x^t) = x^t - \eta \nabla F(x^t) - \eta (\nabla f(x^t) - \nabla F(x^t)) \]

- population-level analysis holds \textit{approximately} if \( x^t \) is independent of \( \{a_l\} \)

a region with well-controlled residual
Back to finite-sample analysis

\[ \mathbf{x}^{t+1} = \mathbf{x}^t - \eta \nabla f(\mathbf{x}^t) = \mathbf{x}^t - \eta \nabla F(\mathbf{x}^t) - \eta (\nabla f(\mathbf{x}^t) - \nabla F(\mathbf{x}^t)) \]

- population-level analysis holds approximately if \( \mathbf{x}^t \) is independent of \( \{a_l\} \)

- key analysis ingredient: show \( \mathbf{x}^t \) is “nearly-independent” of each \( a_l \)

a region with well-controlled residual
Stage 2: local refinement (implicit regularization)

<table>
<thead>
<tr>
<th>iteration complexity</th>
<th>prior theory</th>
<th>our theory</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$O(\frac{n \log \frac{1}{\epsilon}}{\epsilon})$ (Candes et al. '14)</td>
<td>$O(\log \frac{1}{\epsilon}$)</td>
</tr>
</tbody>
</table>
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 $\alpha$-strongly convex and $\beta$-smooth if

\[
0 \preceq \alpha I \preceq \nabla^2 f(x) \preceq \beta I, \quad \forall x
\]
Gradient descent theory revisited

$f$ is said to be $\alpha$-strongly convex and $\beta$-smooth if

\[
0 \leq \alpha I \leq \nabla^2 f(x) \leq \beta I, \quad \forall x
\]

$\ell_2$ error contraction: GD with $\eta = 1/\beta$ obeys

\[
\|x^{t+1} - x^\nabla\|_2 \leq \left(1 - \frac{\alpha}{\beta}\right) \|x^t - x^\nabla\|_2
\]
Gradient descent theory revisited

$f$ is said to be $\alpha$-strongly convex and $\beta$-smooth if

$$0 \preceq \alpha I \preceq \nabla^2 f(x) \preceq \beta I, \quad \forall x$$

\textbf{$l_2$ error contraction:} GD with $\eta = 1/\beta$ obeys

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

- Condition number $\beta/\alpha$ determines rate of convergence
Gradient descent theory revisited

\( f \) is said to be \( \alpha \)-strongly convex and \( \beta \)-smooth if

\[
0 \preceq \alpha I \preceq \nabla^2 f(x) \preceq \beta I, \quad \forall x
\]

\( \ell_2 \) error contraction: GD with \( \eta = 1/\beta \) obeys

\[
\|x^{t+1} - x^\bullet\|_2 \leq \left(1 - \frac{\alpha}{\beta}\right) \|x^t - x^\bullet\|_2
\]

- Condition number \( \beta/\alpha \) determines rate of convergence
- Attains \( \varepsilon \)-accuracy within \( O\left(\frac{\beta}{\alpha} \log \frac{1}{\varepsilon}\right) \) iterations
What does this optimization theory say about GD?

Gaussian designs: \( a_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n), \quad 1 \leq k \leq m \)
What does this optimization theory say about GD?

Gaussian designs: \( a_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n), \quad 1 \leq k \leq m \)

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

\[ \nabla^2 f(x) \succ 0.5I \]
What does this optimization theory say about GD?

\[ \text{Gaussian designs: } a_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n), \quad 1 \leq k \leq m \]

**Finite-sample level** (\(m \approx n \log n\))

\[ \nabla^2 f(x) \succ 0.5I \quad \underbrace{\text{but ill-conditioned}}_{\text{condition number } \asymp n} \quad (\text{even locally}) \]
What does this optimization theory say about GD?

Gaussian designs: \( a_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n), \ 1 \leq k \leq m \)

Finite-sample level \((m \approx n \log n)\)

\[ \nabla^2 f(x) \succ 0.5 I \] but ill-conditioned \( (\text{even locally}) \)

Condition number \( \asymp n \)

Consequence (Candès et al. ’14): WF attains \( \varepsilon \)-accuracy within \( O(n \log \frac{1}{\varepsilon}) \) iterations if \( m \approx n \log n \)
What does this optimization theory say about GD?

Gaussian designs: \( a_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_n), \quad 1 \leq k \leq m \)

Finite-sample level \((m \approx n \log n)\)

\[ \nabla^2 f(x) \succ 0.5 I \quad \text{but ill-conditioned} \quad (\text{even locally}) \]

condition number \(\asymp n\)

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

— 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?

\[ \nabla^2 f(x) = \frac{1}{m} \sum_{k=1}^{3} (a_k^\top x)^2 (a_k a_k^\top - I) \]

• Not sufficiently smooth if \( x \) and \( a_k \) are too close
• \( x \) is incoherent w.r.t. sampling vectors \( \{a_k\} \) (incoherence region)

Prior works suggest enforcing regularization (e.g. truncation, projection, regularized loss) to promote incoherence
A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

\[ \nabla^2 f(x) = \frac{1}{m} \sum_{k=1}^{m} 3(a_k^\top x)^2 a_k a_k^\top - \frac{1}{m} \sum_{k=1}^{m} (a_k^\top x^\perp)^2 a_k a_k^\top \]
A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

\[
\nabla^2 f(x) = \frac{1}{m} \sum_{k=1}^{m} 3 (a_k^\top x)^2 a_k a_k^\top - \frac{1}{m} \sum_{k=1}^{m} (a_k^\top x) a_k a_k^\top
\]

- Not sufficiently smooth if \( x \) and \( a_k \) are too close
A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

- \( x \) is incoherent w.r.t. sampling vectors \( \{a_k\} \) (incoherence region)
A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

- $x$ is incoherent w.r.t. sampling vectors $\{a_k\}$ (incoherence region)
A second look at gradient descent theory

Which local region enjoys both strong convexity and smoothness?

- \( x \) is incoherent w.r.t. sampling vectors \( \{a_k\} \) (incoherence region)

Prior works suggest enforcing regularization (e.g. truncation, projection, regularized loss) to promote incoherence
Aside: truncated WF (Chen, Candès ’15)

Regularize / trim gradients to achieve optimal convergence rate
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness
Our findings: GD is implicitly regularized

- region of local strong convexity + smoothness
Our findings: GD is implicitly regularized

- region of local strong convexity + smoothness
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness
Our findings: GD is implicitly regularized

region of local strong convexity + smoothness

GD implicitly forces iterates to remain incoherent with \( \{a_l\} \)

\[
\max_l |a_l^\top x^t| \lesssim \sqrt{\log n} \|x^t\|_2, \quad \forall t
\]
Our findings: GD is implicitly regularized

- region of local strong convexity + smoothness

GD implicitly forces iterates to remain incoherent with \( \{a_l\} \)

\[
\max_l |a_l^\top x^t| \lesssim \sqrt{\log n} \|x^t\|_2, \quad \forall t
\]

- 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

e.g. introduce leave-one-out iterates \( x^{t,(l)} \) by running GD without \( l \)th sample
Key proof idea: leave-one-out analysis

- Leave-one-out iterate $\mathbf{x}^{t,(l)}$ is independent of $\mathbf{a}_l$
Key proof idea: leave-one-out analysis

- Leave-one-out iterate $\mathbf{x}^{t,(l)}$ is independent of $\mathbf{a}_l$
- Leave-one-out iterate $\mathbf{x}^{t,(l)} \approx$ true iterate $\mathbf{x}^t$
Key proof idea: leave-one-out analysis

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

$\implies \mathbf{x}^t$ is nearly independent of $\mathbf{a}_l$

nearly orthogonal to
Other leave-one-out sequences

\[ A^{\text{sgn}} \quad y = |A^{\text{sgn}} x^l|^2 \]

\[ x^{t, \text{sgn}}: \text{ indep. of sign info of } \{a_{i,1}\} \]

\[ x^{t, \text{sgn}, (l)}: \text{ indep. of both sign info of } \{a_{i,1}\} \text{ and } a_l \]
Saddle-escaping schemes?

Randomly initialized GD never hits saddle points!

\[ \beta_t \]

\[ \alpha_t \]

Global minimizer

saddle points

\( \eta = 0.01 \)

\( \eta = 0.05 \)

\( \eta = 0.1 \)
Other saddle-escaping schemes based on generic landscape analysis

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Iteration Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>trust-region (Sun et al. ’16)</td>
<td>( n^7 + \log \log \frac{1}{\varepsilon} )</td>
</tr>
<tr>
<td>perturbed GD (Jin et al. ’17)</td>
<td>( n^3 + n \log \frac{1}{\varepsilon} )</td>
</tr>
<tr>
<td>perturbed accelerated GD (Jin et al. ’17)</td>
<td>( n^{2.5} + \sqrt{n} \log \frac{1}{\varepsilon} )</td>
</tr>
<tr>
<td>GD (ours) (Chen et al. ’18)</td>
<td>( \log n + \log \frac{1}{\varepsilon} )</td>
</tr>
</tbody>
</table>

Generic optimization theory yields highly suboptimal convergence guarantees.
Outline

- Phase retrieval / solving random quadratic systems of equations
- Matrix completion
Low-rank matrix completion

Given partial samples $\Omega$ of a low-rank matrix $M$, fill in missing entries.

Fig. credit: Candès
Partial theory for unregularized GD

\[
\text{minimize}_X \quad f(X) = \sum_{(j,k) \in \Omega} \left( e_j^\top X X^\top e_k - M_{j,k} \right)^2
\]
Partial theory for unregularized GD

\[
\text{minimize}_X \ f(X) = \sum_{(j,k) \in \Omega} \left( e_j^\top XX^\top e_k - M_{j,k} \right)^2
\]

- GD with random initialization

<table>
<thead>
<tr>
<th>Stage 1: random init $\rightarrow$ local region</th>
<th>Stage 2: local convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Question Mark]</td>
<td>![Thumbs Up]</td>
</tr>
</tbody>
</table>
Partial theory for unregularized GD

\[
\text{minimize}_X \quad f(X) = \sum_{(j,k) \in \Omega} \left( e_j^\top XX^\top e_k - M_{j,k} \right)^2
\]

- GD with random initialization

<table>
<thead>
<tr>
<th>Stage 1: random init $\rightarrow$ local region</th>
<th>Stage 2: local convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="?" alt="Question Mark" />[148x47]</td>
<td><img src="%E2%88%9A" alt="Success" />[148x49]</td>
</tr>
</tbody>
</table>

- GD with spectral initialization

<table>
<thead>
<tr>
<th>Stage 1: spectral initialization</th>
<th>Stage 2: local convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="%E2%88%9A" alt="Success" />[148x49]</td>
<td><img src="%E2%88%9A" alt="Success" />[148x49]</td>
</tr>
</tbody>
</table>
Theoretical guarantees for unregularized GD

Theorem 2 (Matrix completion)

Suppose $M$ is rank-$r$, incoherent and well-conditioned. Vanilla gradient descent with spectral initialization achieves $\varepsilon$-accuracy

- in $O\left(\log \frac{1}{\varepsilon}\right)$ iterations

if step size $\eta \lesssim \frac{1}{\sigma_{\text{max}}(M)}$ and sample size $\gtrsim nr^3 \log^3 n$
Theoretical guarantees for unregularized GD

Theorem 2 (Matrix completion)

Suppose $\mathbf{M}$ is rank-$r$, incoherent and well-conditioned. Vanilla gradient descent with spectral initialization achieves $\varepsilon$-accuracy

- in $O\left(\log \frac{1}{\varepsilon}\right)$ iterations w.r.t. $\| \cdot \|_F$, $\| \cdot \|$, and $\| \cdot \|_{2,\infty}$

if step size $\eta \lesssim \frac{1}{\sigma_{\text{max}}(\mathbf{M})}$ and sample size $\gtrsim nr^3 \log^3 n$
Theoretical guarantees for unregularized GD

Theorem 2 (Matrix completion)

Suppose $M$ is rank-$r$, incoherent and well-conditioned. Vanilla gradient descent with spectral initialization achieves $\varepsilon$-accuracy

- in $O(\log \frac{1}{\varepsilon})$ iterations w.r.t. $\| \cdot \|_F$, $\| \cdot \|$, and $\| \cdot \|_{2,\infty}$

if step size $\eta \lesssim 1/\sigma_{\text{max}}(M)$ and sample size $\gtrsim nr^3 \log^3 n$

- Byproduct: vanilla GD controls entrywise error for noisy MC
  — errors are spread out across all entries
Theoretical guarantees for unregularized GD

**Theorem 2 (Matrix completion)**

Suppose $\mathcal{M}$ is rank-$r$, incoherent and well-conditioned. **Vanilla gradient descent** with **spectral initialization** achieves $\varepsilon$-accuracy

- in $O\left(\log \frac{1}{\varepsilon}\right)$ iterations w.r.t. $\| \cdot \|_F$, $\| \cdot \|$, and $\| \cdot \|_{2,\infty}$ *incoherence*

if step size $\eta \lesssim \frac{1}{\sigma_{\text{max}}(\mathcal{M})}$ and sample size $\gtrsim nr^3 \log^3 n$

- **Byproduct**: vanilla GD controls *entrywise error* for *noisy* MC — errors are spread out across all entries

- **No regularization** is needed
  - in stark contrast to prior works (Keshavan et al. ’10, Sun, Luo ’14, Ge et al. ’16, Chen, Wainwright ’15, Zheng, Lafferty ’16, Chen, Li ’17)
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.