Taming Nonconvexity in Statistical and Reinforcement Learning

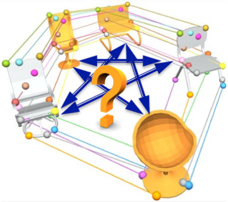
Yuxin Chen

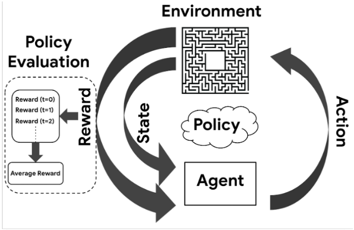
Princeton University

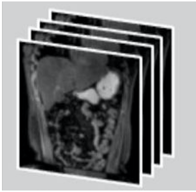
Nonconvex problems are everywhere

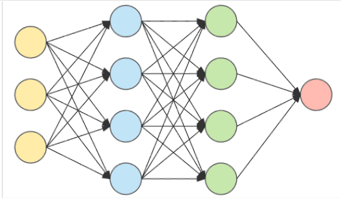
Empirical risk minimization is usually nonconvex

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

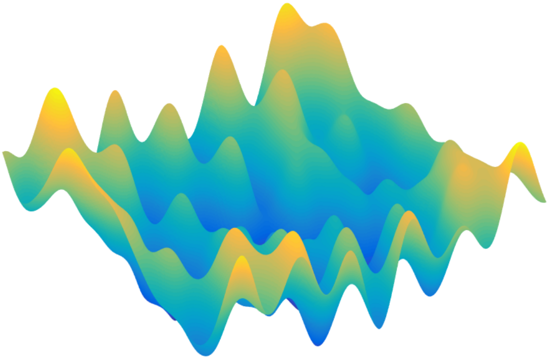








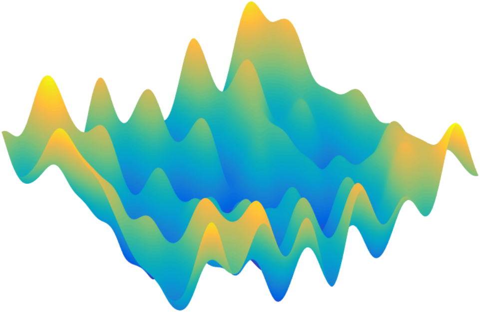
Nonconvex optimization may be super scary

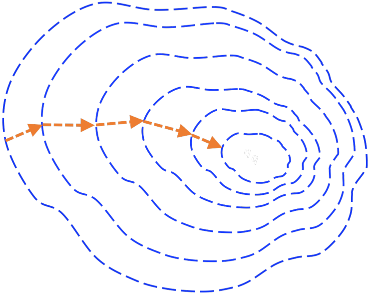


There may be bumps everywhere and exponentially many local optima

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

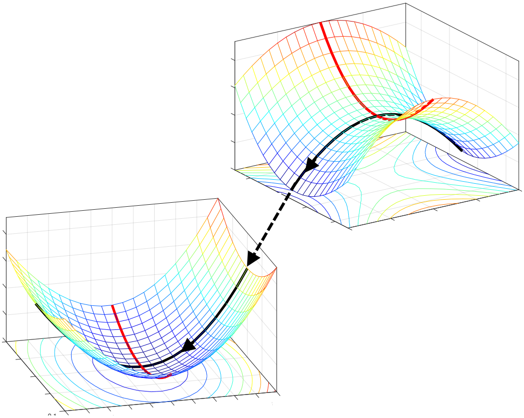
Nonconvex optimization may be super scary



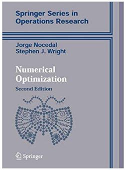


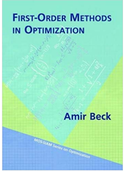
But they are solved on a daily basis via simple algorithms like *(stochastic) gradient descent*

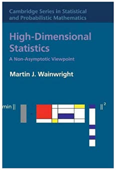
Towards demystifying nonconvex optimization

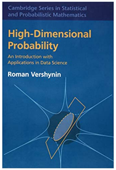


- Characterize optimization landscapes
- Exploit statistical tools to understand finite-sample behavior







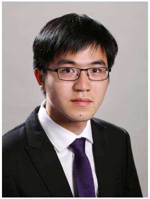
nonconvex optimization

(high-dimensional) statistics

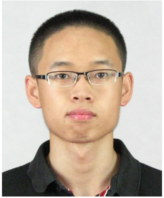
1. Nonconvex statistical learning
   — *an efficient nonconvex algorithm for noisy tensor completion*

2. Nonconvex reinforcement learning
   — *an expotential lower bound for policy gradient methods & an efficient remedy*

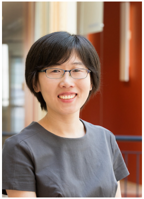
1. Nonconvex optimization for tensor completion



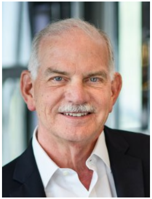
Changxiao Cai
Princeton



Gen Li
Tsinghua



Yuejie Chi
CMU



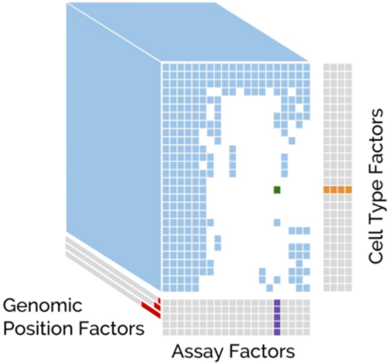
H. Vincent Poor
Princeton

"Nonconvex low-rank tensor completion from noisy data," C. Cai, G. Li, H. Poor, Y. Chen, *Operations Research*, 2021+

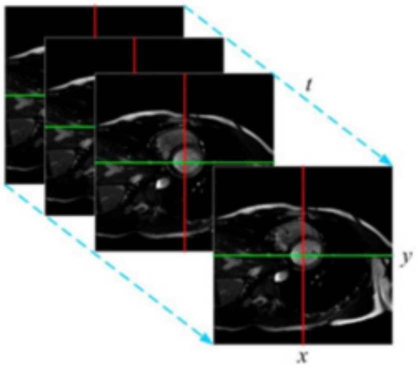
"Subspace estimation from unbalanced and incomplete data matrices: $\ell_{2,\infty}$ statistical guarantees," C. Cai, G. Li, Y. Chi, H. Poor, Y. Chen, *Annals of Statistics*, 2021+

"Uncertainty quantification for nonconvex tensor completion," C. Cai, H. Poor, Y. Chen, *ICML*, 2020

Ubiquity of high-dimensional tensor data

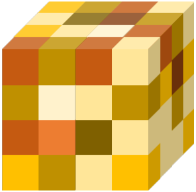


computational genomics
— *fig. credit: Schreiber et al. 19*

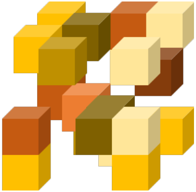


dynamic MRI
— *fig. credit: Liu et al. 17*

Imperfect data acquisition

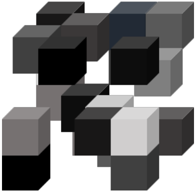


a tensor of interest



missing data

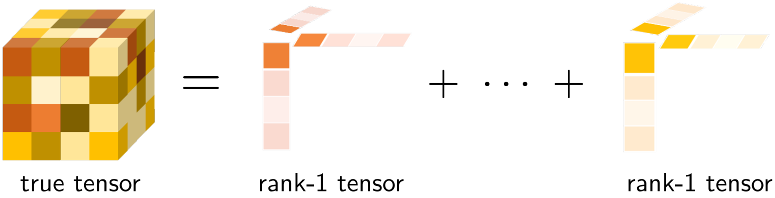
+



noise

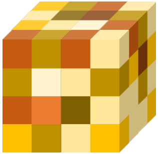
Key to enabling reliable reconstruction from incomplete data

— exploiting **low CP-rank structure**

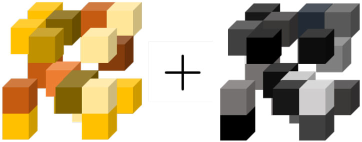


$$\boldsymbol{T}^{\star} = \sum_{i=1}^{r} \boldsymbol{u}_i^{\star} \otimes \boldsymbol{u}_i^{\star} \otimes \boldsymbol{u}_i^{\star}$$

Setup



$\boldsymbol{T}^\star$

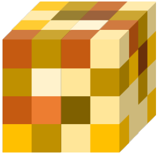


$\boldsymbol{T}$

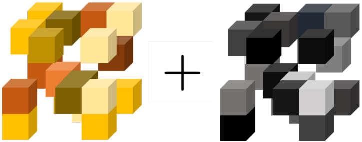
- unknown rank-r tensor $\boldsymbol{T}^\star$:

$$\boldsymbol{T}^\star = \sum_{i=1}^{r} \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \in \mathbb{R}^{d \times d \times d}$$

Setup



$\boldsymbol{T}^\star$



$\boldsymbol{T}$

- unknown rank-r tensor $\boldsymbol{T}^\star$:

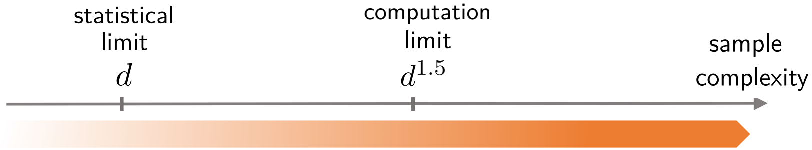
$$\boldsymbol{T}^\star = \sum_{i=1}^{r} \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \in \mathbb{R}^{d\times d\times d}$$

- incomplete & noisy observations over a random sampling set Ω:

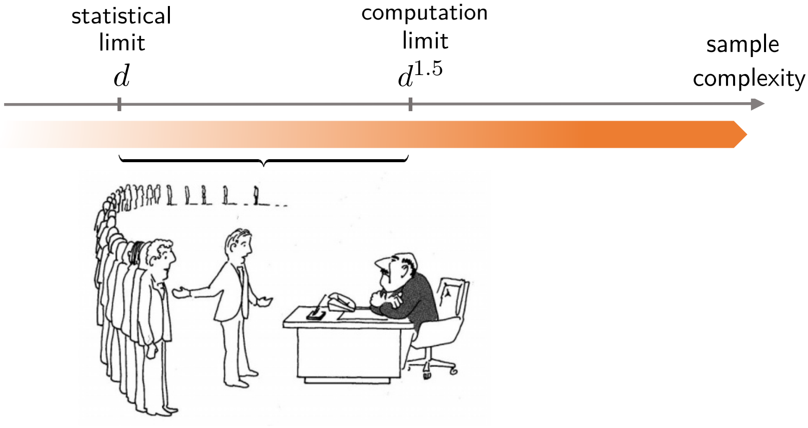
$$T_{i,j,k} = T^\star_{i,j,k} + \text{noise}, \qquad (i,j,k) \in \Omega$$

Goal: recover $\{\boldsymbol{u}_i^\star\}_{i=1}^r$ and $\boldsymbol{T}^\star$

Statistical-computational gap ($r = O(1)$)

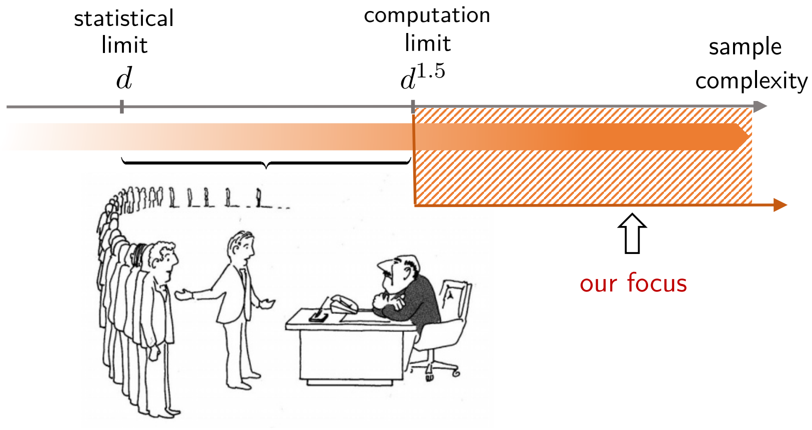


Statistical-computational gap ($r = O(1)$)



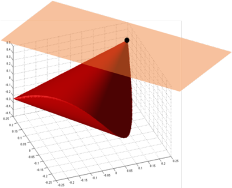
"I can't find an efficient algorithm, but neither can all these people."

Statistical-computational gap ($r = O(1)$)

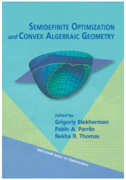


"I can't find an efficient algorithm, but neither can all these people."

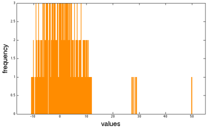
Prior art



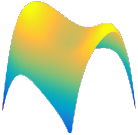
convex relaxation



sum-of-squares hierarchy



spectral methods



nonconvex optimization

- Gandy, Recht, Yamada '11
- Liu, Musialski, Wonka, Ye '12
- Kressner, Steinlechner, Vandereycken '13
- Xu, Hao, Yin, Su '13
- Romera-Paredes, Pontil '13
- Jain, Oh '14
- Huang, Mu, Goldfarb, Wright '15
- Barak, Moitra '16
- Zhang, Aeron '16
- Yuan, Zhang '16
- Montanari, Sun '16
- Kasai, Mishra '16
- Potechin, Steurer '17
- Dong, Yuan, Zhang '17
- Xia, Yuan '19
- Zhang '19
- ...

Prior art ($r = O(1)$)

	algorithm	sample size	comput. cost	recovery type (noiseless)
Yuan, Zhang '16	tensor nuclear norm	d	NP-hard	exact
Xia, Yuan '17	spectral method + GD on manifold	$d^{3/2}$	slow	exact
Montanari, Sun '18	spectral method	$d^{3/2}$	d^3	inexact
Barak, Moitra '16	sum-of-squares	$d^{3/2}$	slow (d^{15})	exact
Potechin et al. '17	sum-of-squares	$d^{3/2}$	slow (d^{10})	exact

Prior art ($r = O(1)$)







	algorithm	sample size	comput. cost	recovery type (noiseless)
Yuan, Zhang '16	tensor nuclear norm	d	NP-hard	exact
Xia, Yuan '17	spectral method + GD on manifold	$d^{3/2}$	slow	exact
Montanari, Sun '18	spectral method	$d^{3/2}$	d^3	inexact
Barak, Moitra '16	sum-of-squares	$d^{3/2}$	slow (d^{15})	exact
Potechin et al. '17	sum-of-squares	$d^{3/2}$	slow (d^{10})	exact

	algorithm	ℓ_2 error (noisy)	ℓ_∞ error (noisy)
Xia, Yuan, Zhang '17	spectral method + tensor power method	suboptimal	n/a
Barak, Moitra '16	sum-of-squares	suboptimal	n/a

Can we design an algorithm that is simultaneously
sample-efficient, computationally fast, & minimax-optimal?

A nonconvex least squares formulation

$$\underset{\boldsymbol{U}=[\boldsymbol{u}_1,\cdots,\boldsymbol{u}_r]\in\mathbb{R}^{d\times r}}{\text{minimize}} \; f(\boldsymbol{U}) := \underbrace{\sum_{(i,j,k)\in\Omega} \left\{ \Big(\sum\nolimits_{s=1}^{r} \boldsymbol{u}_s^{\otimes 3} \Big)_{i,j,k} - T_{i,j,k} \right\}^2}_{\text{squared loss over observed entries}}$$

A nonconvex least squares formulation

$$\underset{\boldsymbol{U}=[\boldsymbol{u}_1,\cdots,\boldsymbol{u}_r]\in\mathbb{R}^{d\times r}}{\text{minimize}} \; f(\boldsymbol{U}) := \underbrace{\sum_{(i,j,k)\in\Omega} \Big\{ \Big(\textstyle\sum_{s=1}^{r} \boldsymbol{u}_s^{\otimes 3} \Big)_{i,j,k} - T_{i,j,k} \Big\}^2}_{\text{squared loss over observed entries}}$$

- **pros:** statistically efficient *if we can find global solutions*

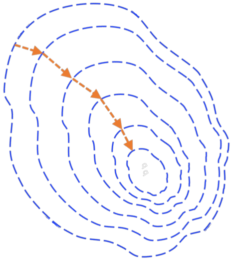
A nonconvex least squares formulation

$$\underset{\boldsymbol{U}=[\boldsymbol{u}_1,\cdots,\boldsymbol{u}_r]\in\mathbb{R}^{d\times r}}{\text{minimize}}\ f(\boldsymbol{U}) := \underbrace{\sum_{(i,j,k)\in\Omega}\Big\{\Big(\sum\nolimits_{s=1}^{r}\boldsymbol{u}_s^{\otimes 3}\Big)_{i,j,k} - T_{i,j,k}\Big\}^2}_{\text{squared loss over observed entries}}$$

- **pros:** statistically efficient *if we can find global solutions*
- **cons:** highly nonconvex $\longrightarrow$ computationally challenging

Gradient descent (GD) with random initialization?

$$\underset{\boldsymbol{U}=[\boldsymbol{u}_1,\cdots,\boldsymbol{u}_r]\in\mathbb{R}^{d\times r}}{\text{minimize}} \; f(\boldsymbol{U}) := \sum_{(i,j,k)\in\Omega}\left\{\left(\sum\nolimits_{s=1}^{r}\boldsymbol{u}_s^{\otimes 3}\right)_{i,j,k} - T_{i,j,k}\right\}^2$$



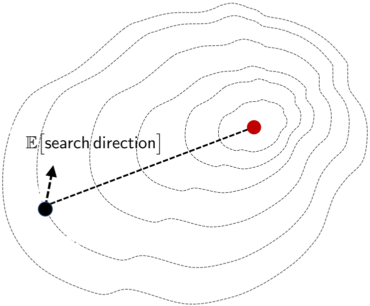
- **initialize** $\boldsymbol{U}^0$ randomly
- **gradient descent:** for $t = 0, 1, \cdots,$

$$\boldsymbol{U}^{t+1} = \boldsymbol{U}^t - \eta_t \nabla f(\boldsymbol{U}^t)$$

— *succeeds for phase retrieval (Chen et al. '18)*

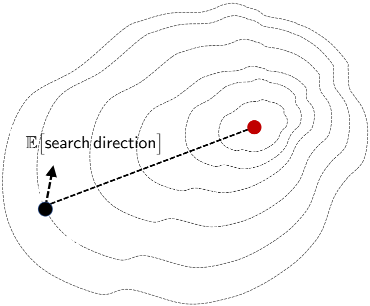
A negative conjecture

Randomly initialized GD does NOT work unless sample size $> \ d^2$



A negative conjecture

Randomly initialized GD does NOT work unless sample size $> \; d^2$

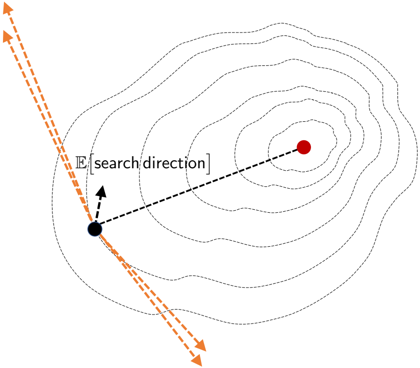


When sample size $\asymp \; d^{1.5}$:

- $\mathbb{E}\big[$search direction$\big]$ is desirable

A negative conjecture

Randomly initialized GD does NOT work unless sample size $> \ d^2$

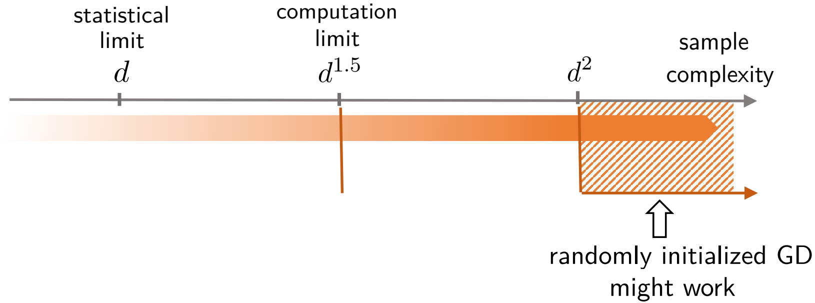


When sample size $\asymp \ d^{1.5}$:

- $\mathbb{E}[\text{search direction}]$ is desirable
- **issue:** variance $\gtrsim \sqrt{d}\,\text{mean}^2$

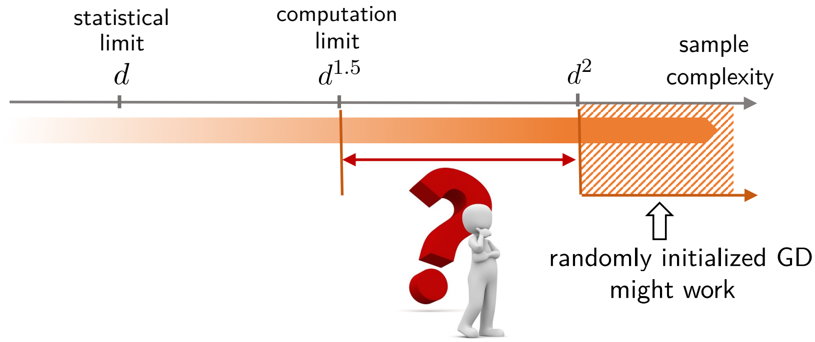
A negative conjecture

Randomly initialized GD does NOT work unless sample size $> \ d^2$

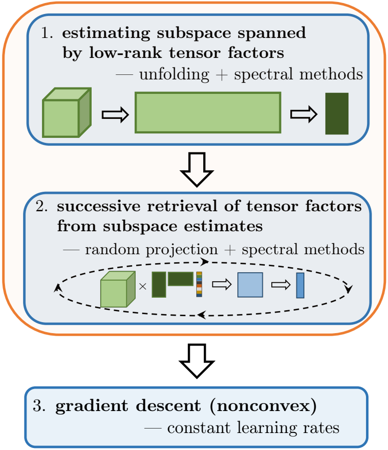


A negative conjecture

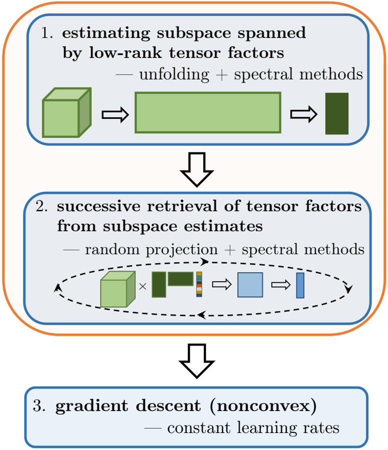
Randomly initialized GD does NOT work unless sample size $> \;\; d^2$



Our proposal: a two-stage nonconvex algorithm



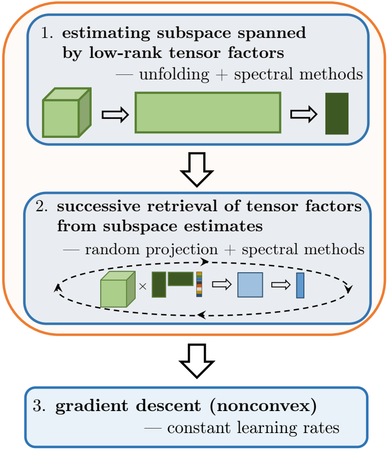
Our proposal: a two-stage nonconvex algorithm



1. initialization: $\boldsymbol{U}^0$

- estimate span$\{\boldsymbol{u}_i^\star\}$ via spectral method
- disentangle individual factors $\{\boldsymbol{u}_i^\star\}$ from subspace estimate

Our proposal: a two-stage nonconvex algorithm



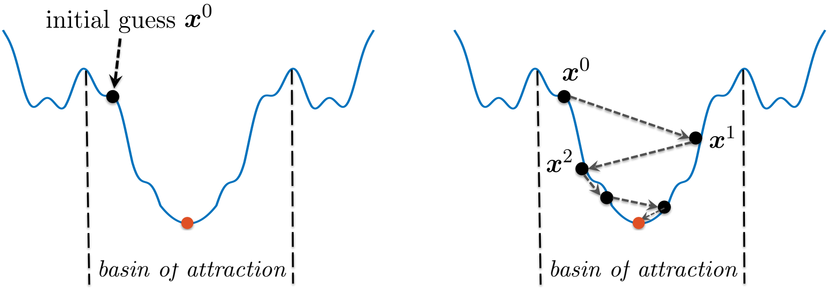
1. initialization: $\boldsymbol{U}^0$

- estimate span$\{\boldsymbol{u}_i^\star\}$ via spectral method
- disentangle individual factors $\{\boldsymbol{u}_i^\star\}$ from subspace estimate

2. gradient descent: for $t = 0, 1, \cdots$

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

Rationale of two-stage approach



1. initialize within $\underbrace{\text{a local basin sufficiently close to global min}}_{\text{(restricted) strongly convex}}$

2. iterative refinement

A bit more details about initialization

Step 1.1: estimate $\mathrm{span}\{\boldsymbol{u}_i^\star\}_{1\le i\le r} \quad \longrightarrow \quad \boldsymbol{U}_{\mathsf{sub}}$

- matricizition: $\boldsymbol{A} = \mathrm{unfold}(\boldsymbol{T})$
- estimate rank-r subspace of $\mathcal{P}_{\mathsf{off\text{-}diag}}(\boldsymbol{A}\boldsymbol{A}^\top)$ (diagonal deletion)

A bit more details about initialization

Step 1.1: estimate span$\{\boldsymbol{u}_i^\star\}_{1\le i\le r} \quad \longrightarrow \quad \boldsymbol{U}_{\mathsf{sub}}$

- matricizition: $\boldsymbol{A} = \mathsf{unfold}(\boldsymbol{T})$
- estimate rank-r subspace of $\mathcal{P}_{\mathsf{off\text{-}diag}}(\boldsymbol{A}\boldsymbol{A}^\top)$ (diagonal deletion)

Step 1.2: retrieve tensor factors from subspace estimate

- generate a random vector $\boldsymbol{g}$ from $\boldsymbol{U}_{\mathsf{sub}}$
- compute leading eigenvector of $\boldsymbol{T}\otimes\boldsymbol{g} = \underbrace{\sum_i \langle \boldsymbol{u}_i^\star, \boldsymbol{g}\rangle \boldsymbol{u}_i^\star \boldsymbol{u}_i^{\star\top} + \text{noise}}_{\text{find the } \boldsymbol{u}_i^\star \text{ most aligned with } \boldsymbol{g}}$
- repeat ...

Assumptions

$$\boldsymbol{T}^\star = \sum_{i=1}^{r} \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \in \mathbb{R}^{d\times d\times d}$$

- **random sampling**: each entry is observed independently with prob. p

Assumptions

$$\boldsymbol{T}^\star = \sum_{i=1}^{r} \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \in \mathbb{R}^{d\times d\times d}$$

- **random sampling**: each entry is observed independently with prob. p
- **random noise**: independent zero-mean sub-Gaussian noise with variance $O(\sigma^2)$

Assumptions

$$\boldsymbol{T}^\star = \sum_{i=1}^{r} \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \in \mathbb{R}^{d\times d\times d}$$

- **random sampling**: each entry is observed independently with prob. p
- **random noise**: independent zero-mean sub-Gaussian noise with variance $O(\sigma^2)$
- **ground truth**: low-rank $\big(r = O(1)\big)$, well-conditioned, incoherent $\big(\{\boldsymbol{u}_i^\star\}$ are de-localized and not aligned$\big)$

ℓ_2 and ℓ_∞ theoretical guarantees

Theorem 1 (Cai, Li, Poor, Chen '19)

There exists some constant $\rho < 1$ *and some permutation matrix* $\mathbf{\Pi} \in \mathbb{R}^{r \times r}$ *s.t. with high prob., the* t*-th iterate satisfies*

$$\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\mathrm{F}}$$

$$\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\mathrm{F}}$$

ℓ_2 and ℓ_∞ theoretical guarantees

Theorem 1 (Cai, Li, Poor, Chen '19)

There exists some constant $\rho < 1$ *and some permutation matrix* $\mathbf{\Pi} \in \mathbb{R}^{r \times r}$ *s.t. with high prob., the* t*-th iterate satisfies*

$$\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\mathrm{F}}$$

$$\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\mathrm{F}}$$

$$\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\infty}$$

$$\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\infty}$$

provided that sample size $\gtrsim d^{1.5} \mathrm{poly}\log(d)$

ℓ_2 and ℓ_∞ theoretical guarantees

Theorem 1 (Cai, Li, Poor, Chen '19)

There exists some constant $\rho < 1$ and some permutation matrix $\mathbf{\Pi} \in \mathbb{R}^{r \times r}$ s.t. with high prob., the t-th iterate satisfies

$$\left\|\boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star\right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\|\boldsymbol{U}^\star\right\|_{\mathrm{F}}$$

$$\left\|\boldsymbol{T}^t - \boldsymbol{T}^\star\right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\|\boldsymbol{T}^\star\right\|_{\mathrm{F}}$$

$$\left\|\boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star\right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\|\boldsymbol{U}^\star\right\|_{\infty}$$

$$\left\|\boldsymbol{T}^t - \boldsymbol{T}^\star\right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\|\boldsymbol{T}^\star\right\|_{\infty}$$

provided that sample size $\gtrsim d^{1.5} \mathrm{poly}\log(d)$

- linear/geometric convergence $\longrightarrow$ linear-time algorithm

ℓ_2 and ℓ_∞ theoretical guarantees

Theorem 1 (Cai, Li, Poor, Chen '19)

There exists some constant $\rho < 1$ *and some permutation matrix* $\mathbf{\Pi} \in \mathbb{R}^{r \times r}$ *s.t. with high prob., the* t*-th iterate satisfies*

$$\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\mathrm{F}}$$

$$\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\mathrm{F}}$$

$$\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\infty}$$

$$\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\infty}$$

provided that sample size $\gtrsim d^{1.5} \mathrm{poly}\log(d)$

- near-optimal sample complexity (among poly-time algorithms)

ℓ_2 and ℓ_∞ theoretical guarantees

Theorem 1 (Cai, Li, Poor, Chen '19)

There exists some constant $\rho < 1$ *and some permutation matrix* $\mathbf{\Pi} \in \mathbb{R}^{r \times r}$ *s.t. with high prob., the* t*-th iterate satisfies*

$$\left\|\boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star\right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \|\boldsymbol{U}^\star\|_{\mathrm{F}}$$

$$\left\|\boldsymbol{T}^t - \boldsymbol{T}^\star\right\|_{\mathrm{F}} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \|\boldsymbol{T}^\star\|_{\mathrm{F}}$$

$$\left\|\boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star\right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \|\boldsymbol{U}^\star\|_{\infty}$$

$$\left\|\boldsymbol{T}^t - \boldsymbol{T}^\star\right\|_{\infty} \lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \|\boldsymbol{T}^\star\|_{\infty}$$

provided that sample size $\gtrsim d^{1.5} \mathrm{poly}\log(d)$

- near-optimal statistical accuracy (both ℓ_2 and ℓ_∞)

ℓ_2 and ℓ_∞ theoretical guarantees

Theorem 1 (Cai, Li, Poor, Chen '19)

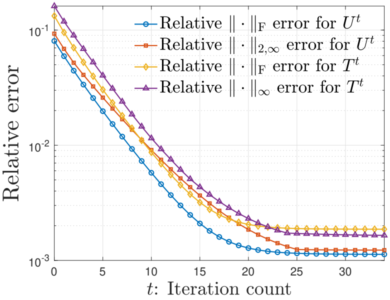
There exists some constant $\rho < 1$ *and some permutation matrix* $\mathbf{\Pi} \in \mathbb{R}^{r \times r}$ *s.t. with high prob., the* t*-th iterate satisfies*

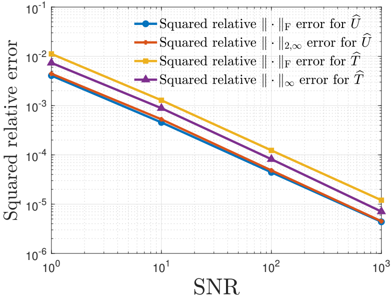
$$\begin{aligned}
\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\mathrm{F}} &\lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\mathrm{F}} \\
\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\mathrm{F}} &\lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\mathrm{F}} \\
\left\| \boldsymbol{U}^t \mathbf{\Pi} - \boldsymbol{U}^\star \right\|_{\infty} &\lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{U}^\star \right\|_{\infty} \\
\left\| \boldsymbol{T}^t - \boldsymbol{T}^\star \right\|_{\infty} &\lesssim \left(\rho^t + \sigma\sqrt{d/p}\right) \left\| \boldsymbol{T}^\star \right\|_{\infty}
\end{aligned}$$

provided that sample size $\gtrsim d^{1.5} \mathrm{poly}\log(d)$

- *no need of* sample splitting
  $\longrightarrow$ can handle complicated stat dependency across iterations

Numerical experiments





$$d = 100,\ r = 4,\ p = 0.1$$

Key proof idea: leave-one-out decoupling

Leave out a small amount of randomness and re-run the algorithm

Key proof idea: leave-one-out decoupling

Leave out a small amount of randomness and re-run the algorithm

- 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
- Ma, Wang, Chi, Chen '17
- Chen, Chi, Fan, Ma '18
- Ding, Chen '18
- Dong, Shi '18
- Chen, Liu, Li '19
- Chen, Fan, Ma, Yan '19
- Pananjady, Wainwright '19
- Ling '20
- Chen, Fan, Ma, Yan '20
- Agarwal, Kakade, Yang '20
- Abbe, Fan, Wang '20
- Li, Wei, Chi, Gu, Chen '20

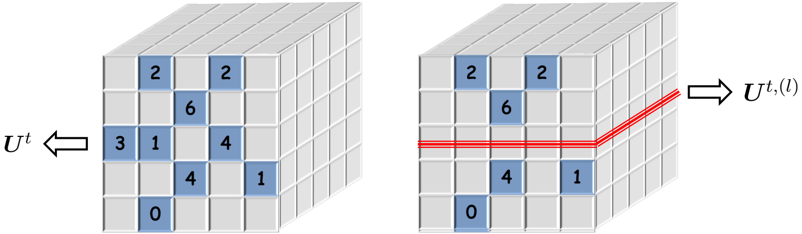
Foundations and Trends® in Machine Learning

Spectral Methods for Data Science: A Statistical Perspective

Suggested Citation: Yuxin Chen, Yuejie Chi, Jianqing Fan and Cong Ma (2020), "Spectral Methods for Data Science: A Statistical Perspective", Foundations and Trends® in

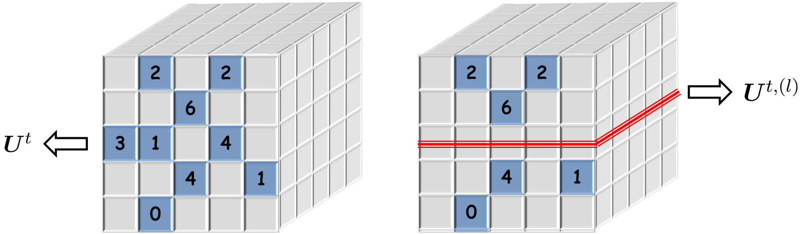
Key proof idea: leave-one-out decoupling

For each $1 \le l \le d$, generate leave-one-out auxiliary iterates $\{\boldsymbol{U}^{t,(l)}\}$ by replacing l^{th} slice with true values



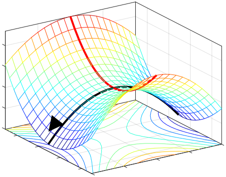
Key proof idea: leave-one-out decoupling

For each $1 \le l \le d$, generate leave-one-out auxiliary iterates $\{\boldsymbol{U}^{t,(l)}\}$ by replacing l^{th} slice with true values



- exploit partial statistical independence
- exploit leave-one-out stability
- enable optimal ℓ_∞ error control

Summary of Part 1



nonconvex
optimization

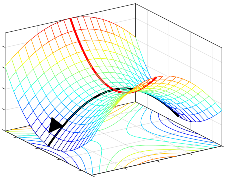
optimal estimation guarantees

linear-time algorithm

minimal sample size

fine-grained uncertainty quantification

Summary of Part 1



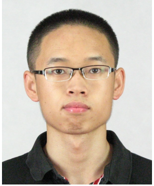
nonconvex optimization

- optimal estimation guarantees
- linear-time algorithm
- minimal sample size
- fine-grained uncertainty quantification

phase retrieval | matrix completion | ranking | blind deconvolution

joint image alignment | mixture models | reinforcement learning

2: Nonconvex optimization in reinforcement learning

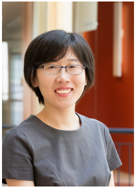


Gen Li
Tsinghua

Shicong Cen
CMU

Chen Cheng
Stanford

Yuting Wei
CMU

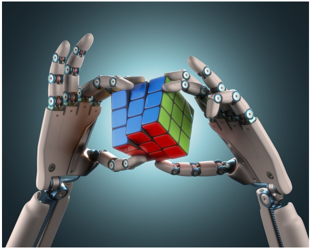


Yuejie Chi
CMU

"Softmax policy gradient methods can take exponential time to converge," G. Li, Y. Wei, Y. Chi, Y. Gu, Y. Chen, arxiv:2102.11270, 2021

"Fast global convergence of natural policy gradient methods with entropy regularization," S. Cen, C. Cheng, Y. Chen, Y. Wei, Y. Chi, under revision, *Operations Research*, 2020

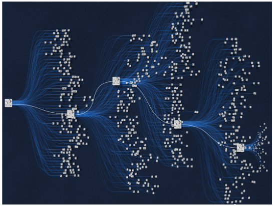
Recent successes in reinforcement learning (RL)



Policy optimization: a major contributor to recent RL advances

RL challenges

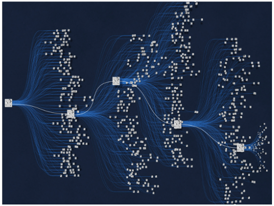
In RL, an agent learns by interacting with an unknown environment



- enormous state and action space
- delayed rewards
- credit assignments
- nonconvexity everywhere

RL challenges

In RL, an agent learns by interacting with an unknown environment

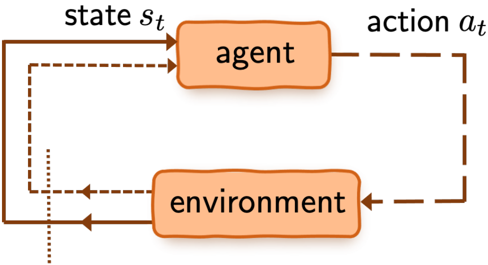


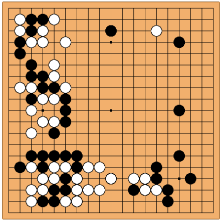
- enormous state and action space
- delayed rewards
- credit assignments
- nonconvexity everywhere

How to enable scalable and guaranteed RL despite nonconvexity?

Backgrounds: policy optimization for MDPs

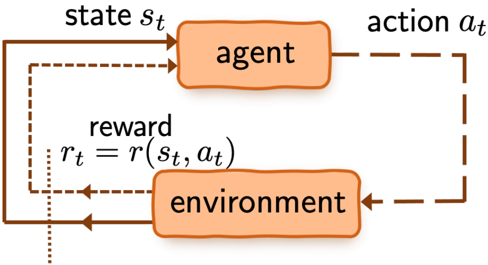
Markov decision process (MDP)

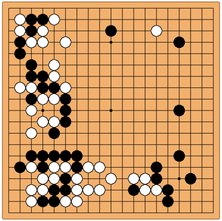




- $\mathcal{S}$: state space
- $\mathcal{A}$: action space

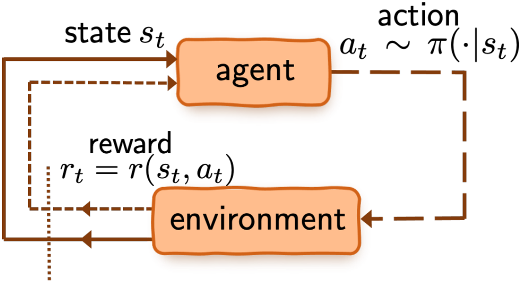
Markov decision process (MDP)

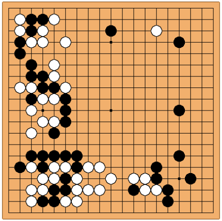




- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward

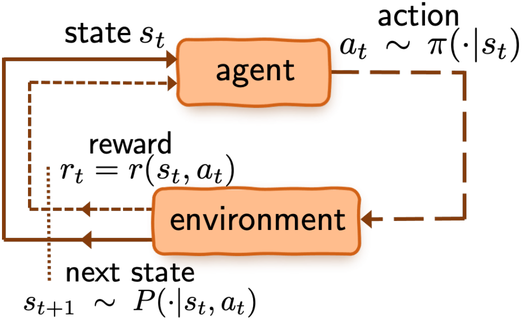
Markov decision process (MDP)

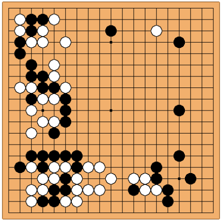




- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot|s)$: policy (or action selection rule)

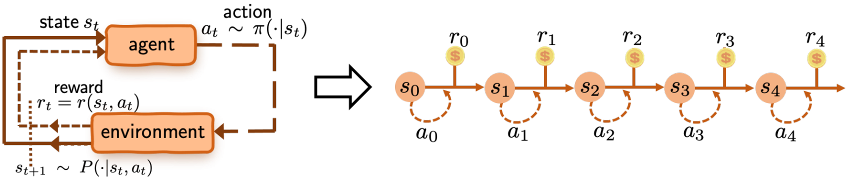
Markov decision process (MDP)





- $\mathcal{S}$: state space
- $\mathcal{A}$: action space
- $r(s, a) \in [0, 1]$: immediate reward
- $\pi(\cdot|s)$: policy (or action selection rule)
- $P(\cdot|s, a)$: transition probabilities

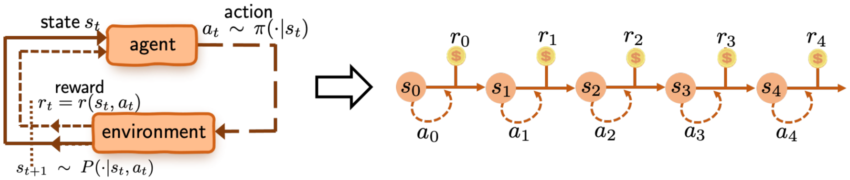
Value function of policy π



cumulative discounted reward: $$V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s\right], s \in \mathcal{S}$$

- expectation is over randomness of MDP & policy π

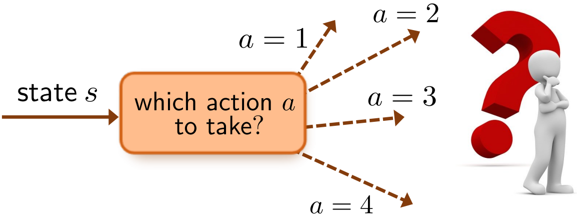
Value function of policy π



cumulative discounted reward: $$V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s\right], s \in \mathcal{S}$$

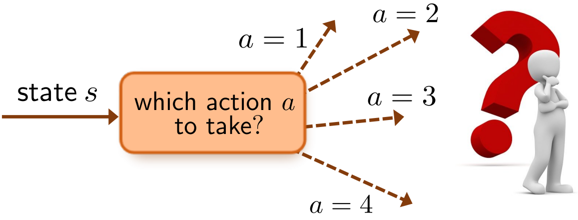
- expectation is over randomness of MDP & policy π
- $\gamma \in [0, 1)$: discount factor
  - take $\gamma \to 1$ to approximate long-horizon MDPs
  - effective horizon: $\frac{1}{1-\gamma}$

Optimal policy and optimal value



- **goal:** find optimal policy $\pi^\star$ that maximizes value functions
- optimal value function: $V^\star(s) := \max_\pi V^\pi(s)$ for all $s \in \mathcal{S}$

Optimal policy and optimal value



- **goal:** find optimal policy $\pi^\star$ that maximizes value functions
- optimal value function: $V^\star(s) := \max_\pi V^\pi(s)$ for all $s \in \mathcal{S}$

How to accomplish it via nonconvex optimization algorithms?

Policy optimization

Given state distribution $s \sim \rho$
(e.g. uniform)

$$\boxed{\max_{\pi} \; V^{\pi}(\rho) := \mathop{\mathbb{E}}_{s \sim \rho}\left[V^{\pi}(s)\right]}$$

Policy optimization

Given state distribution $s \sim \rho$
(e.g. uniform)

$$\boxed{\max_{\pi} \; V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho}\left[V^{\pi}(s)\right]} \xrightarrow{\text{parameterize}} \boxed{\max_{\theta} \; V^{\pi_\theta}(\rho) := \mathbb{E}_{s \sim \rho}\left[V^{\pi_\theta}(s)\right]}$$

Policy optimization

Given state distribution $s \sim \rho$
(e.g. uniform)

$$\max_{\pi} \; V^{\pi}(\rho) := \mathop{\mathbb{E}}_{s \sim \rho}\left[V^{\pi}(s)\right]$$

parameterize $\Longrightarrow$

$$\max_{\theta} \; V^{\pi_\theta}(\rho) := \mathop{\mathbb{E}}_{s \sim \rho}\left[V^{\pi_\theta}(s)\right]$$

$$\pi_\theta(a|s) = \frac{\exp(\theta(s,a))}{\sum_a \exp(\theta(s,a))}$$

softmax parameterization

Policy optimization

Given state distribution $s \sim \rho$
(e.g. uniform)

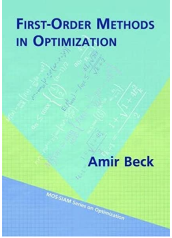
$$\max_{\pi} \; V^{\pi}(\rho) := \mathop{\mathbb{E}}_{s\sim\rho}\left[V^{\pi}(s)\right]$$

parameterize $\Longrightarrow$

$$\max_{\theta} \; V^{\pi_\theta}(\rho) := \mathop{\mathbb{E}}_{s\sim\rho}\left[V^{\pi_\theta}(s)\right]$$

$$\pi_\theta(a|s) = \frac{\exp(\theta(s,a))}{\sum_a \exp(\theta(s,a))}$$

softmax parameterization



Policy optimization

Given state distribution $s \sim \rho$
(e.g. uniform)

$$\max_{\pi} \; V^{\pi}(\rho) := \mathop{\mathbb{E}}_{s\sim\rho}\left[V^{\pi}(s)\right] \quad \xrightarrow{\text{parameterize}} \quad \max_{\theta} \; V^{\pi_\theta}(\rho) := \mathop{\mathbb{E}}_{s\sim\rho}\left[V^{\pi_\theta}(s)\right]$$

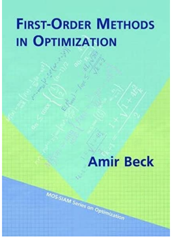
$$\pi_\theta(a|s) = \frac{\exp(\theta(s,a))}{\sum_a \exp(\theta(s,a))}$$

softmax parameterization

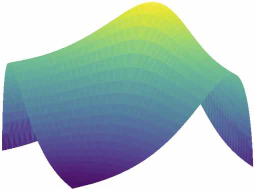
Policy gradient method (Sutton et al. '00)

$$\theta^{(t+1)} = \theta^{(t)} + \eta \nabla_\theta V^{(t)}(\rho), \qquad t = 0, 1, \cdots$$

- η: learning rate

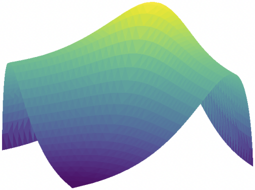


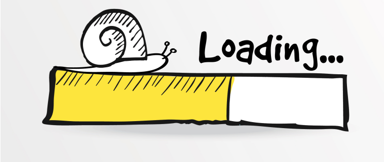
Does policy gradient (PG) method converge?



- (Agarwal et al. '19) Softmax PG converges to global opt as $t \to \infty$

Does policy gradient (PG) method converge?

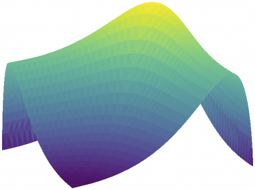


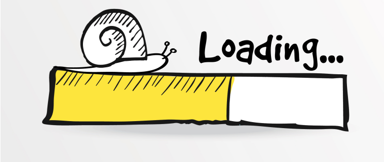


- (Agarwal et al. '19) Softmax PG converges to global opt as $t \to \infty$

However, "asymptotic convergence" might mean "taking forever"

Does policy gradient (PG) method converge?



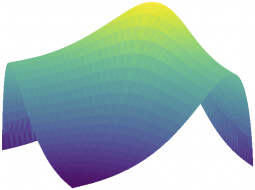


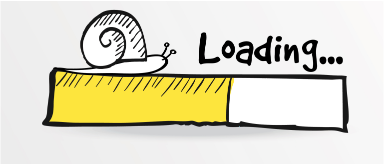
- (Agarwal et al. '19) Softmax PG converges to global opt as $t \to \infty$
- (Mei et al. '20) Softmax PG converges to global opt in

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

However, "asymptotic convergence" might mean "taking forever"

Does policy gradient (PG) method converge?





- (Agarwal et al. '19) Softmax PG converges to global opt as $t \to \infty$
- (Mei et al. '20) Softmax PG converges to global opt in

$$c(|\mathcal{S}|, |\mathcal{A}|, \tfrac{1}{1-\gamma}, \cdots)\, O(\tfrac{1}{\varepsilon}) \text{ iterations}$$

However, "asymptotic convergence" might mean "taking forever"

A negative message

Theorem 2 (Li, Wei, Chi, Chen ’21)

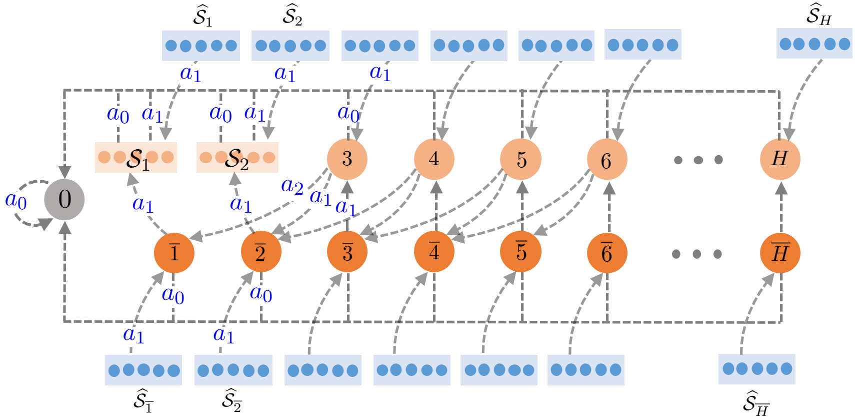
There exists an MDP s.t. it takes softmax PG at least

$$\frac{1}{\eta}\,|\mathcal{S}|^{1.5^{\Theta(\frac{1}{1-\gamma})}} \text{ iterations}$$

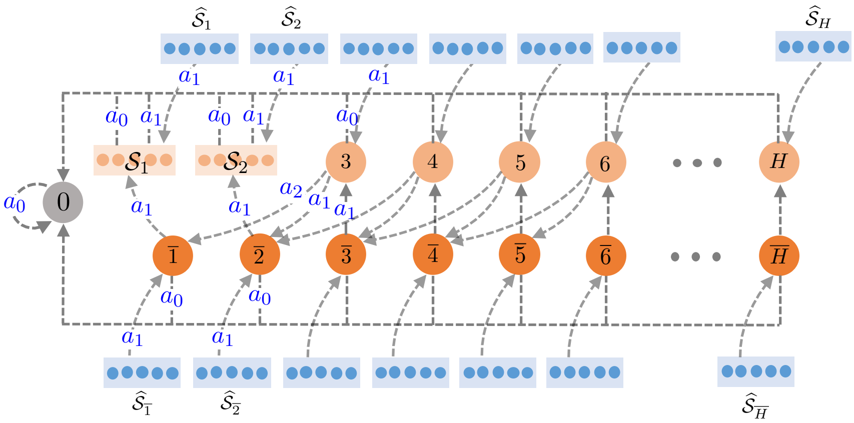
to achieve $\|V^{(t)} - V^{\star}\|_{\infty} \leq 1/2$ *(even with infinite samples)*

- Softmax PG method can take exponential time to converge (in problems w/ large state space & long effective horizon)!

MDP construction for our lower bound



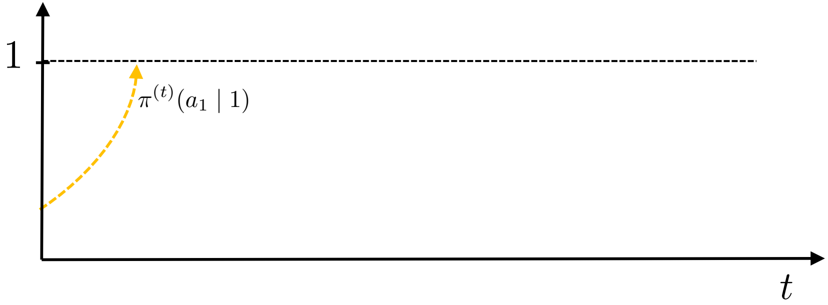
MDP construction for our lower bound



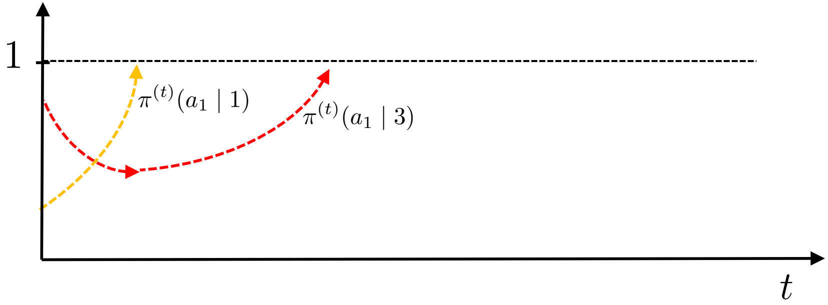
Key design ingredients: for $3 \leq s \leq H \asymp \frac{1}{1-\gamma}$,

- a_1 is optimal action
- delayed rewards
- $\pi^{(t)}(a_1|s)$ keeps decreasing until $\pi^{(t)}(a_1|s-2) \approx 1$

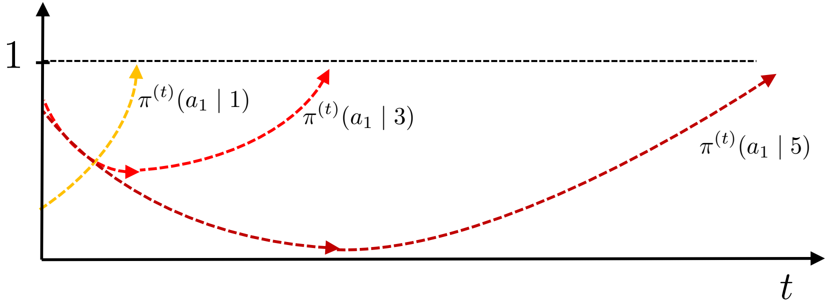
What is happening in our constructed MDP?



What is happening in our constructed MDP?

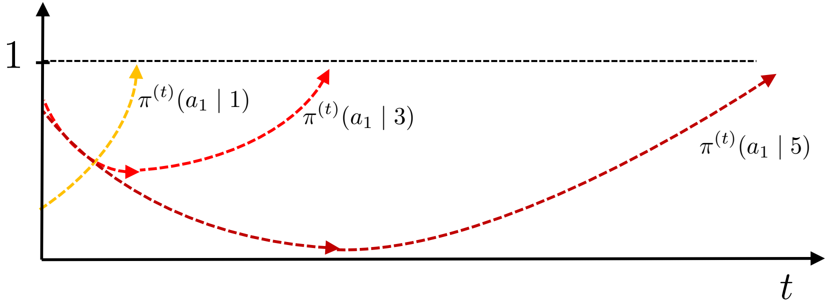


What is happening in our constructed MDP?



observation: convergence time for state s grows geometrically as $s \uparrow$

What is happening in our constructed MDP?



observation: convergence time for state s grows geometrically as $s \uparrow$

$$\text{convergence-time}(s) \gtrsim \big(\text{convergence-time}(s-2)\big)^{1.5}$$

Booster 1: entropy regularization

accelerate convergence by regularizing objective function

$$V_\tau^\pi(s_0) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \big(r_t - \tau \log \pi(a_t|s_t)\big) \,\Big|\, s_0\right]$$

Booster 1: entropy regularization

accelerate convergence by regularizing objective function

$$
\begin{aligned}
V_\tau^\pi(s_0) &:= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \big(r_t - \tau \log \pi(a_t|s_t)\big) \,\Big|\, s_0\right] \\
&= V^\pi(s) + \frac{\tau}{1-\gamma} \mathop{\mathbb{E}}_{s\sim d_s^\pi} \Big[\underbrace{-\sum_a \pi(a|s)\log\pi(a|s)}_{\text{Shannon entropy}} \,\Big|\, s_0\Big]
\end{aligned}
$$

- τ: regularization parameter
- d_s^π: certain marginal distribution

Booster 1: entropy regularization

accelerate convergence by regularizing objective function

$$
\begin{aligned}
V_\tau^\pi(s_0) &:= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \big(r_t - \tau \log \pi(a_t|s_t)\big) \,\Big|\, s_0\right] \\
&= V^\pi(s) + \frac{\tau}{1-\gamma} \mathop{\mathbb{E}}_{s\sim d_s^\pi} \Big[\underbrace{-\sum_a \pi(a|s) \log \pi(a|s)}_{\text{Shannon entropy}} \,\Big|\, s_0\Big]
\end{aligned}
$$

- τ: regularization parameter
- d_s^π: certain marginal distribution

entropy-regularized value maximization

$$\text{maximize}_\theta \quad V_\tau^{\pi_\theta}(\rho) := \mathbb{E}_{s\sim\rho}\big[V_\tau^{\pi_\theta}(s)\big]$$

Our negative message remains . . .

Theorem 3 (Li, Wei, Chi, Chen '21)

There is an MDP s.t. it takes entropy-regularized softmax PG at least

$$\min\left\{\exp\left(\Theta\left(\frac{1}{\varepsilon}\right)\right),\ \frac{1}{\eta}|\mathcal{S}|^{2^{\Theta(\frac{1}{1-\gamma})}}\right\} \textit{ iterations}$$

to achieve $\|V^{(t)} - V^{\star}\|_{\infty} \leq \varepsilon$ *(even with infinite samples)*

- Softmax PG method with entropy regularization can still take exponential time to converge!

Our negative message remains ...

Theorem 3 (Li, Wei, Chi, Chen '21)

There is an MDP s.t. it takes entropy-regularized softmax PG at least

$$\min\left\{\exp\left(\Theta\left(\frac{1}{\varepsilon}\right)\right),\ \frac{1}{\eta}|\mathcal{S}|^{2^{\Theta\left(\frac{1}{1-\gamma}\right)}}\right\}\ \textit{iterations}$$

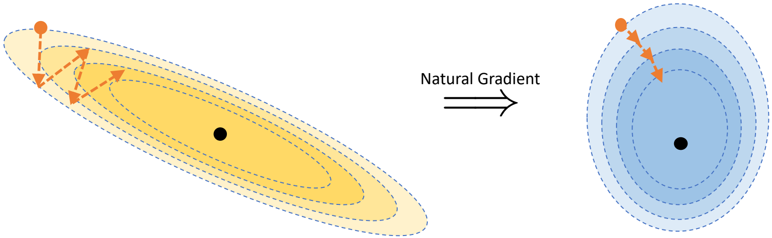
to achieve $\|V^{(t)} - V^{\star}\|_{\infty} \leq \varepsilon$ *(even with infinite samples)*

- Softmax PG method with entropy regularization can still take exponential time to converge!
- (Mei et al. '20) entropy-regularized softmax PG converges in

$$c(|\mathcal{S}|, |\mathcal{A}|, \tfrac{1}{1-\gamma}, \cdots)\, O(\tfrac{1}{\varepsilon}) \text{ iterations}$$

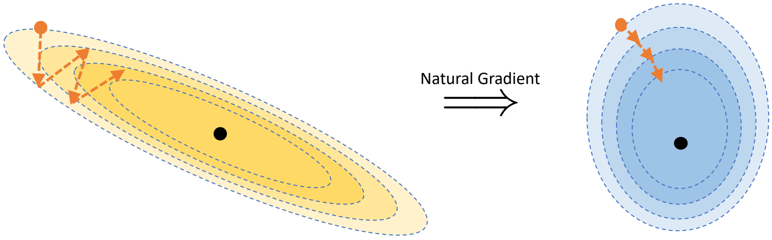
Booster 2: natural policy gradient (NPG)

precondition gradients to improve search directions ...



Booster 2: natural policy gradient (NPG)

precondition gradients to improve search directions ...



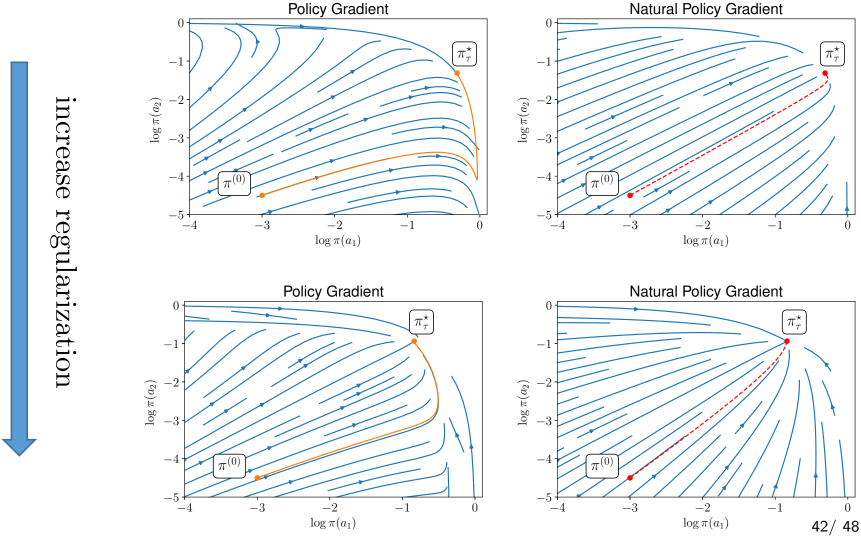
NPG method (Kakade '02)

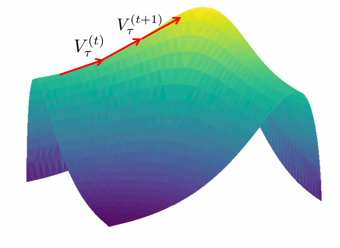
$$\theta^{(t+1)} = \theta^{(t)} + \eta (\mathcal{F}_\rho^\theta)^\dagger \nabla_\theta V_\tau^{(t)}(\rho), \qquad t = 0, 1, \cdots$$

- $\mathcal{F}_\rho^\theta := \mathbb{E}\left[\left(\nabla_\theta \log \pi_\theta(a|s)\right)\left(\nabla_\theta \log \pi_\theta(a|s)\right)^\top\right]$: Fisher info

Entropy-regularized natural gradient helps!

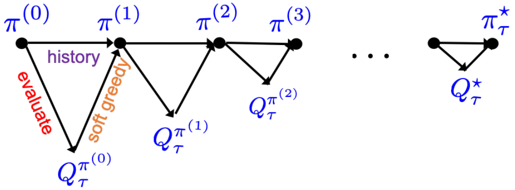
A toy bandit example: 3 arms with rewards 1, 0.9 and 0.1





How to characterize the efficiency of entropy-regularized NPG in tabular settings?

Linear convergence with exact gradients



exact access to gradients:

Theorem 4 (Cen, Cheng, Chen, Wei, Chi '20)

If $\eta \leq (1-\gamma)/\tau$ *and* $\tau \leq \frac{(1-\gamma)\varepsilon}{4\log|\mathcal{A}|}$, *entropy-regularized NPG achieves*

$$\|V^\star - V^{(t+1)}\|_\infty \leq C_1\gamma\,(1-\eta\tau)^t + \varepsilon/2, \quad t = 0, 1, \cdots$$

Implications: iteration complexity

number of iterations needed to reach $\|V^\star - V^{(t)}\|_\infty \le \varepsilon$:

- **general learning rates ($0 < \eta < \frac{1-\gamma}{\tau}$):**

$$\frac{1}{\eta\tau} \log\left(\frac{2C_1\gamma}{\varepsilon}\right)$$

- **optimal choice ($\eta = \frac{1-\gamma}{\tau}$):**

$$\frac{1}{1-\gamma} \log\left(\frac{2C_1\gamma}{\varepsilon}\right)$$

Implications: iteration complexity

number of iterations needed to reach $\|V^{\star} - V^{(t)}\|_{\infty} \le \varepsilon$:

- **general learning rates ($0 < \eta < \frac{1-\gamma}{\tau}$):**

$$\frac{1}{\eta\tau} \log\left(\frac{2C_1\gamma}{\varepsilon}\right)$$

- **optimal choice ($\eta = \frac{1-\gamma}{\tau}$):**

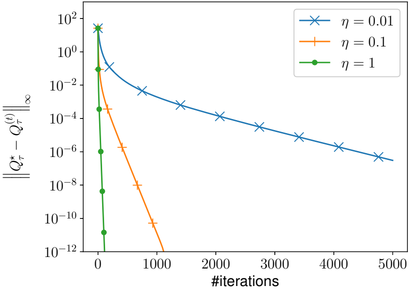
$$\frac{1}{1-\gamma} \log\left(\frac{2C_1\gamma}{\varepsilon}\right)$$

Nearly dimension-free global linear convergence!

Regularized NPG vs. unregularized NPG

regularized NPG

$\tau = 0.001$

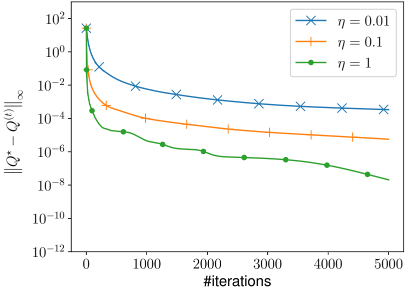


linear rate: $\frac{1}{\eta\tau}\log\left(\frac{1}{\varepsilon}\right)$

Ours

unregularized NPG

$\tau = 0$



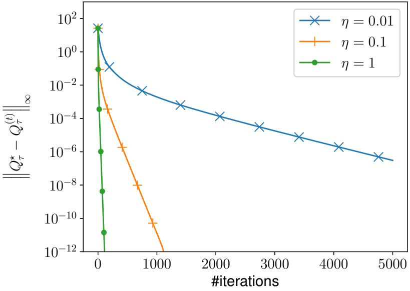
sublinear rate: $\frac{1}{\min\{\eta,(1-\gamma)^2\}\varepsilon}$

(Agarwal et al. '19)

Regularized NPG vs. unregularized NPG

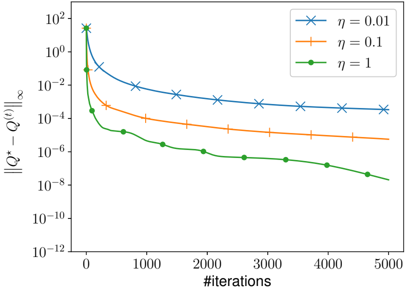
regularized NPG

$\tau = 0.001$



unregularized NPG

$\tau = 0$



linear rate: $\frac{1}{\eta\tau}\log\left(\frac{1}{\varepsilon}\right)$

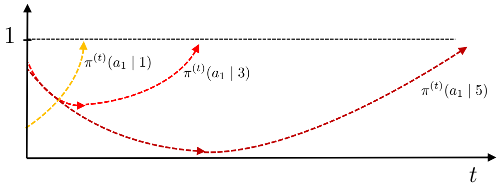
Ours

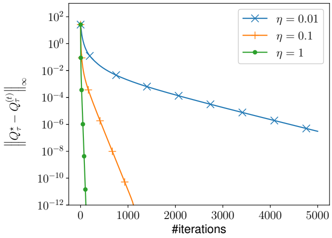
sublinear rate: $\frac{1}{\min\{\eta,(1-\gamma)^2\}\varepsilon}$

(Agarwal et al. '19)

Entropy regularization enables faster convergence!

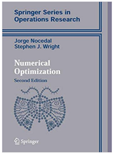
Summary of Part 2

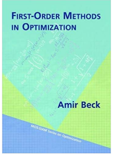




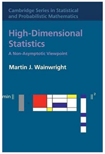
- Softmax policy gradient can take exponential time to converge
- Entropy regularization & natural gradients help!

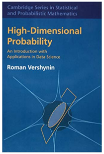
Concluding remarks





nonconvex optimization





(high-dimensional) statistics