Algorithmic regularization in asymmetric overparameterized matrix sensing

Dominik Stöger

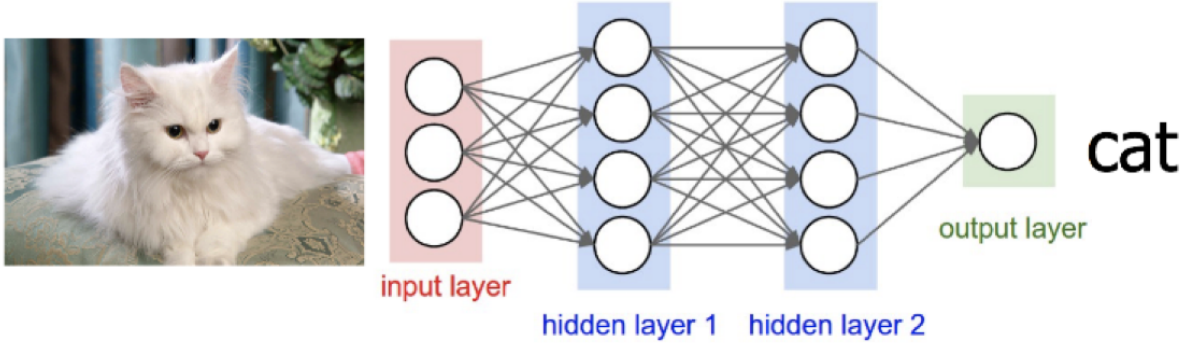
KU Eichstätt-Ingolstadt

October 13, 2023
LMCRC Workshop on Mathematical Information Science

Mahdi Soltanolkotabi (USC)

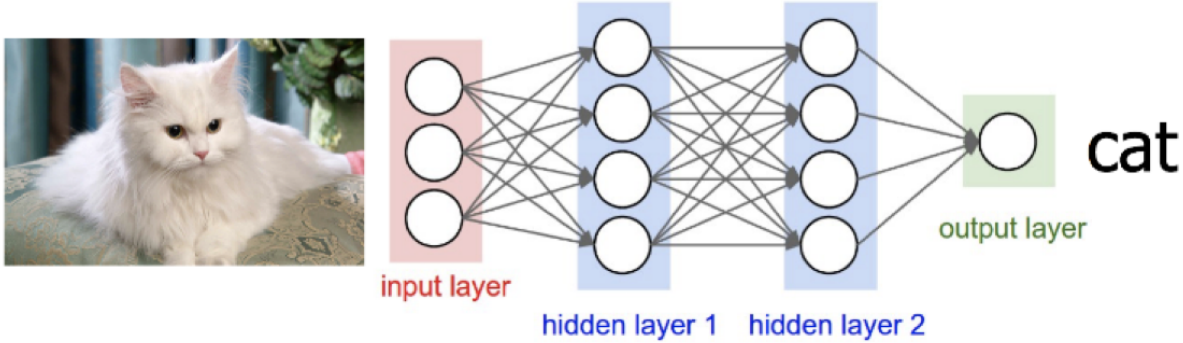
Changzhi Xie (USC)

Statistical surprise in modern machine learning

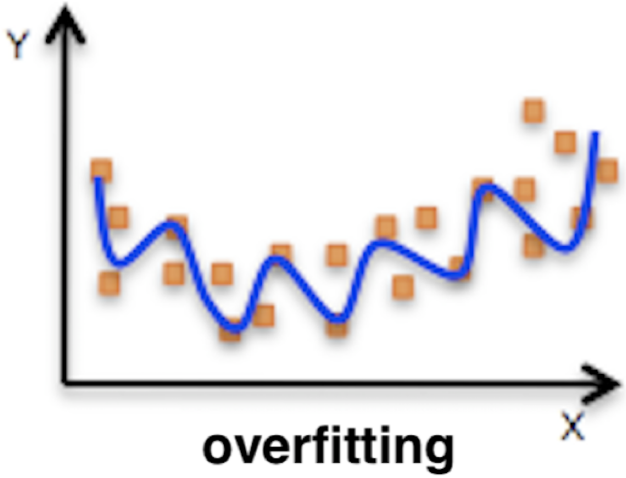


Overparameterization:
Number of parameters of neural network $\gg$ number of training points

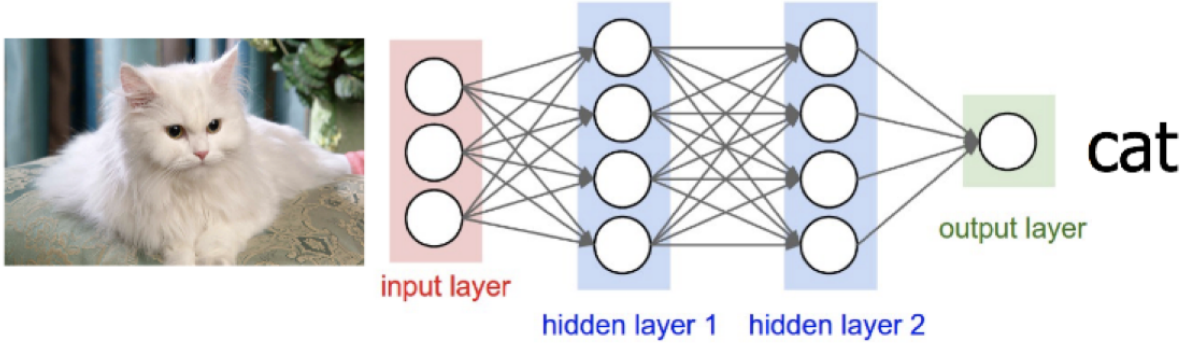
Statistical surprise in modern machine learning



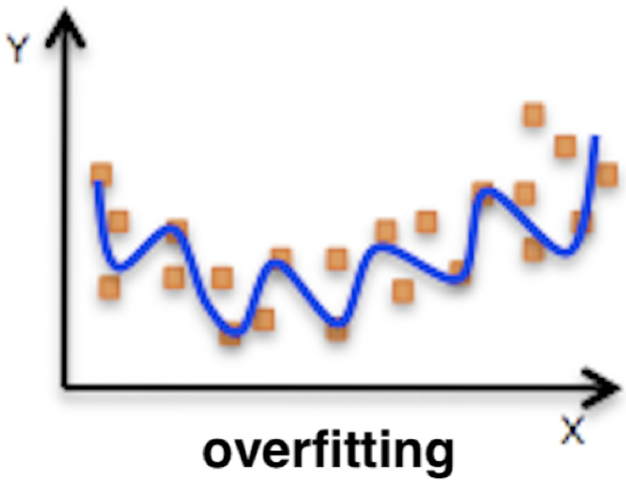
Overparameterization:
Number of parameters of neural network $\gg$ number of training points

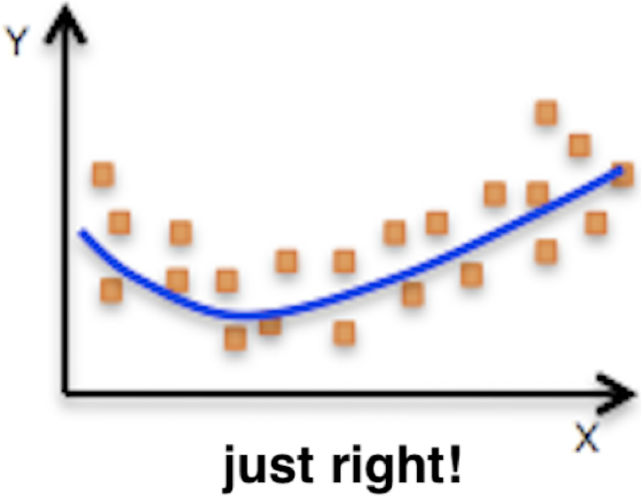


Statistical surprise in modern machine learning



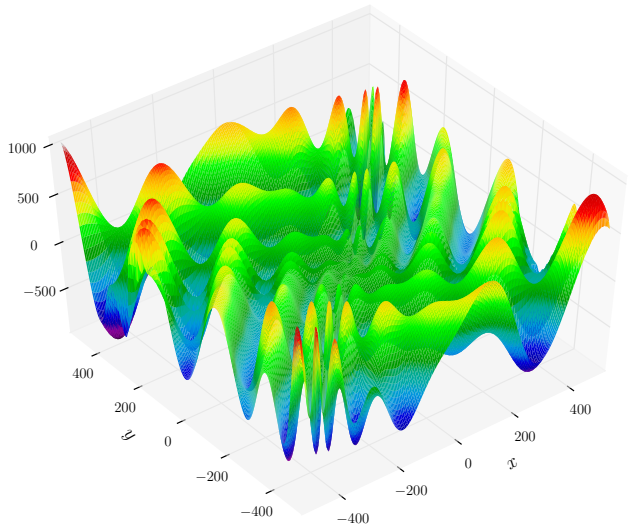
Overparameterization:
Number of parameters of neural network $\gg$ number of training points





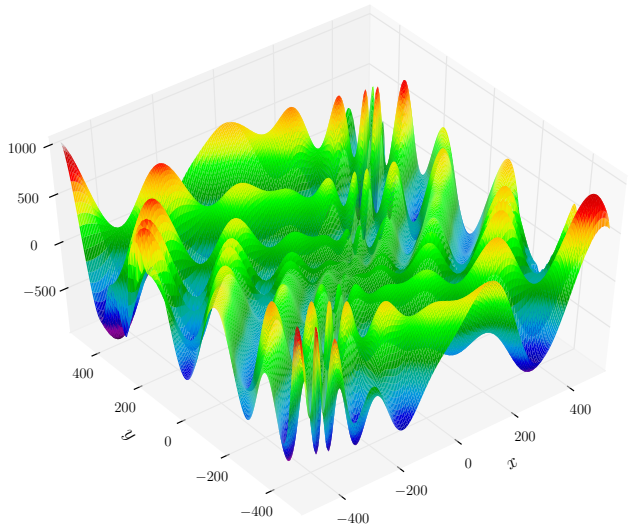
Optimization surprise in modern machine learning

- Models trained by optimizing a non-convex loss function f via (stochastic) gradient descent
- Potentially, many saddle points and local minima...



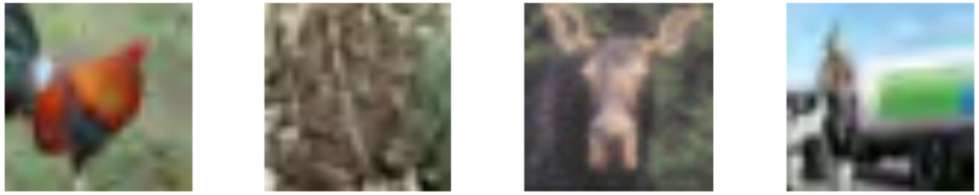
Optimization surprise in modern machine learning

- Models trained by optimizing a non-convex loss function f via (stochastic) gradient descent
- Potentially, many saddle points and local minima...



- (Stochastic) gradient descent can find a global optimum in many scenarios!
- **Empirical observation:** More overparameterization often makes the optimization task easier!

Experiment on CIFAR-10 dataset (Zhang et al.)

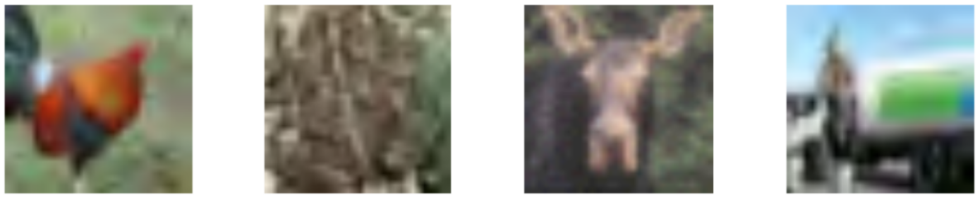


CIFAR10

- $n = 50,000$ training samples
- 10 different classes
- p: number of parameters of the trained model

[1] See Zhang, "Deep learning and structured data", Ph.D. thesis

Experiment on CIFAR-10 dataset (Zhang et al.)



CIFAR10

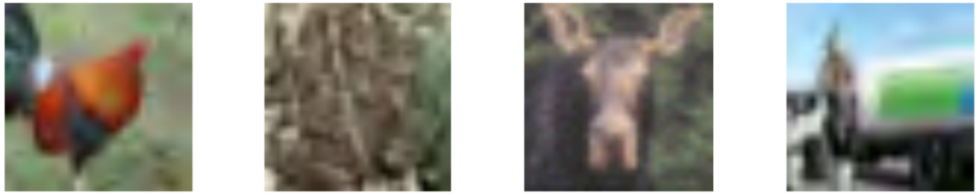
- $n = 50,000$ training samples
- 10 different classes
- p: number of parameters of the trained model

Architecture	p/n ratio	Training Accuracy	Test Accuracy
MLP 1×512	24	100%	51.51 %
Alexnet	28	100%	76.07 %
Inception	33	100%	85.75 %
Wide Resnet	179	100%	88.21 %

Good generalization even when no (explicit) regularization is used!

[1] See Zhang, "Deep learning and structured data", Ph.D. thesis

Experiment on CIFAR-10 dataset (Zhang et al.)



CIFAR10

- $n = 50,000$ training samples
- 10 different classes
- p: number of parameters of the trained model

Architecture	p/n ratio	Training Accuracy	Test Accuracy
MLP 1×512	24	100%	51.51 %
Alexnet	28	100%	76.07 %
Inception	33	100%	85.75 %
Wide Resnet	179	100%	88.21 %

Good generalization even when no (explicit) regularization is used!
No overfitting!

[1] See Zhang, "Deep learning and structured data", Ph.D. thesis

Why does overparameterization seem to help both with optimization as well as with statistical performance? Why does the model not overfit?

Prelude: Overparameterized linear least squares

- Solve

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} g(\boldsymbol{\theta}) = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2,$$

  where $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ has full rank and $n < p$

- Use gradient descent, i.e.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \mu \nabla g(\boldsymbol{\theta}_t)$$

Prelude: Overparameterized linear least squares

- Solve

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} g(\boldsymbol{\theta}) = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2,$$

  where $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ has full rank and $n < p$
- Use gradient descent, i.e.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \mu \nabla g(\boldsymbol{\theta}_t)$$

- since $n < p$ and $\boldsymbol{X}$ has full rank, infinitely many minimizers $\boldsymbol{\theta}$ with $g(\boldsymbol{\theta}) = 0$

Prelude: Overparameterized linear least squares

- Solve

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} g(\boldsymbol{\theta}) = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2,$$

  where $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ has full rank and $n < p$
- Use gradient descent, i.e.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \mu \nabla g(\boldsymbol{\theta}_t)$$

- since $n < p$ and $\boldsymbol{X}$ has full rank, infinitely many minimizers $\boldsymbol{\theta}$ with $g(\boldsymbol{\theta}) = 0$

Lemma (folklore)

For small enough step size and initialization $\boldsymbol{\theta}_0 = \mathbf{0}$, gradient descent converges to the unique solution of

$$\min \|\boldsymbol{\theta}\|_2 \qquad \text{subject to} \qquad \boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta}.$$

The implicit regularization phenomenon

The least squares example suggests the following explanation for the success of gradient descent in overparameterized learning problems (without explicit regularization).

Implicit regularization hypothesis

Among all solutions, gradient descent converges to the one which minimizes a certain complexity parameter (such as ℓ_2-norm, ℓ_1-norm, rank,...).

The implicit regularization phenomenon

The least squares example suggests the following explanation for the success of gradient descent in overparameterized learning problems (without explicit regularization).

Implicit regularization hypothesis

Among all solutions, gradient descent converges to the one which minimizes a certain complexity parameter (such as ℓ_2-norm, ℓ_1-norm, rank,...).

In this talk: Understanding overparameterization in low-rank matrix learning problems

Low-rank matrix learning problem

- $\boldsymbol{X}_* \in \mathbb{R}^{n \times n}$ unknown matrix of rank $r_\star \ll n$
- Given m observations of the form

$$y_i = \langle \boldsymbol{A}_i, \boldsymbol{X}_* \rangle := \mathrm{Tr}\left(\boldsymbol{A}_i^\top \boldsymbol{X}_*\right)$$

- measurement matrices $\{\boldsymbol{A}_i\}_{i=1}^m$ have i.i.d. Gaussian entries
- **Goal:** Learn low-rank matrix $\boldsymbol{X}_*$ given measurements $(y_i)_{i=1}^m$

Low-rank matrix learning problem

- ***Non-convex*** objective

$$\mathcal{L}(\boldsymbol{V}, \boldsymbol{W}) = \frac{1}{2m} \sum_{i=1}^{m} \left(y_i - \langle \boldsymbol{A}_i, \boldsymbol{V}\boldsymbol{W}^\top \rangle \right)^2,$$

where $\boldsymbol{V}, \boldsymbol{W} \in \mathbb{R}^{n \times k}$ for $k \geq r_\star$

Low-rank matrix learning problem

- ***Non-convex*** objective

$$\mathcal{L}(\boldsymbol{V}, \boldsymbol{W}) = \frac{1}{2m} \sum_{i=1}^{m} \left(y_i - \langle \boldsymbol{A}_i, \boldsymbol{V}\boldsymbol{W}^\top \rangle \right)^2,$$

where $\boldsymbol{V}, \boldsymbol{W} \in \mathbb{R}^{n \times k}$ for $k \geq r_\star$

- ***Goal*** **:** Find $\boldsymbol{V}$ and $\boldsymbol{W}$, which minimize **test loss**

$$\mathbb{E}_{\boldsymbol{A}} \left(\langle \boldsymbol{A}, \boldsymbol{X}_* \rangle - \langle \boldsymbol{A}, \boldsymbol{V}\boldsymbol{W}^\top \rangle \right)^2 = \| \boldsymbol{X}_* - \boldsymbol{V}\boldsymbol{W}^\top \|_F^2$$

Minimum is achieved for $\boldsymbol{X}_* = \boldsymbol{V}\boldsymbol{W}^\top$

Overparameterization in low-rank matrix recovery

- Recall that $\boldsymbol{V}, \boldsymbol{W} \in \mathbb{R}^{n \times k}$, $\boldsymbol{X}_* \in \mathbb{R}^{n \times n}$ with rank $r_\star$, and m number of observations
- **Overparameterized regime:**

$$rk \gg m \gg r_\star n$$

Overparameterization in low-rank matrix recovery

- Recall that $\boldsymbol{V}, \boldsymbol{W} \in \mathbb{R}^{n \times k}$, $\boldsymbol{X}_* \in \mathbb{R}^{n \times n}$ with rank $r_\star$, and m number of observations
- **Overparameterized regime:**

$$rk \gg m \gg r_\star n$$

- There exist infinitely many $(\boldsymbol{V}, \boldsymbol{W})$ such that $\mathcal{L}(\boldsymbol{V}, \boldsymbol{W}) = 0$, but $\|\boldsymbol{X}_* - \boldsymbol{V}\boldsymbol{W}^\top\|_F$ is arbitrarily large

Overparameterization in low-rank matrix recovery

- Recall that $\boldsymbol{V}, \boldsymbol{W} \in \mathbb{R}^{n \times k}$, $\boldsymbol{X}_* \in \mathbb{R}^{n \times n}$ with rank $r_\star$, and m number of observations
- **Overparameterized regime:**

$$rk \gg m \gg r_\star n$$

- There exist infinitely many $(\boldsymbol{V}, \boldsymbol{W})$ such that $\mathcal{L}(\boldsymbol{V}, \boldsymbol{W}) = 0$, but $\|\boldsymbol{X}_* - \boldsymbol{V}\boldsymbol{W}^\top\|_F$ is arbitrarily large

What if we just run gradient descent on the non-convex objective $\mathcal{L}$ without explicit regularization?

Observation (following Gunasekar et al.)

- $\overline{\boldsymbol{V}}, \overline{\boldsymbol{W}} \in \mathbb{R}^{n \times k}$ shape matrices with i.i.d. Gaussian entries $\mathcal{N}(0, 1)$
- $\alpha > 0$ scale of initialization
- Set $\boldsymbol{V_0} := \alpha \overline{\boldsymbol{V}}$ and $\boldsymbol{W_0} := \alpha \overline{\boldsymbol{W}}$

Observation (following Gunasekar et al.)

- $\overline{\boldsymbol{V}}, \overline{\boldsymbol{W}} \in \mathbb{R}^{n \times k}$ shape matrices with i.i.d. Gaussian entries $\mathcal{N}(0,1)$
- $\alpha > 0$ scale of initialization
- Set $\boldsymbol{V_0} := \alpha \overline{\boldsymbol{V}}$ and $\boldsymbol{W_0} := \alpha \overline{\boldsymbol{W}}$
- Minimize loss function f via gradient descent

$$\boldsymbol{V_{t+1}} = \boldsymbol{V_t} - \mu \nabla_{\boldsymbol{V}} \mathcal{L}(\boldsymbol{V_t})$$
$$\boldsymbol{W_{t+1}} = \boldsymbol{W_t} - \mu \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W_t})$$

Observation (following Gunasekar et al.)

- $\overline{\boldsymbol{V}}, \overline{\boldsymbol{W}} \in \mathbb{R}^{n \times k}$ shape matrices with i.i.d. Gaussian entries $\mathcal{N}(0,1)$
- $\alpha > 0$ scale of initialization
- Set $\boldsymbol{V_0} := \alpha \overline{\boldsymbol{V}}$ and $\boldsymbol{W_0} := \alpha \overline{\boldsymbol{W}}$
- Minimize loss function f via gradient descent

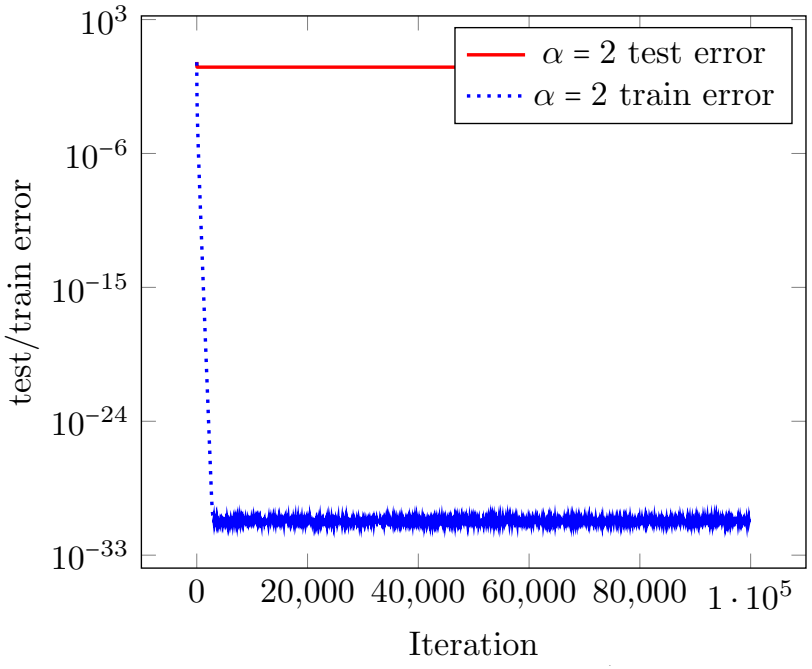
$$\boldsymbol{V_{t+1}} = \boldsymbol{V_t} - \mu \nabla_{\boldsymbol{V}} \mathcal{L}(\boldsymbol{V_t})$$
$$\boldsymbol{W_{t+1}} = \boldsymbol{W_t} - \mu \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W_t})$$

- Empirical observation by Gunasekar et al ('18):

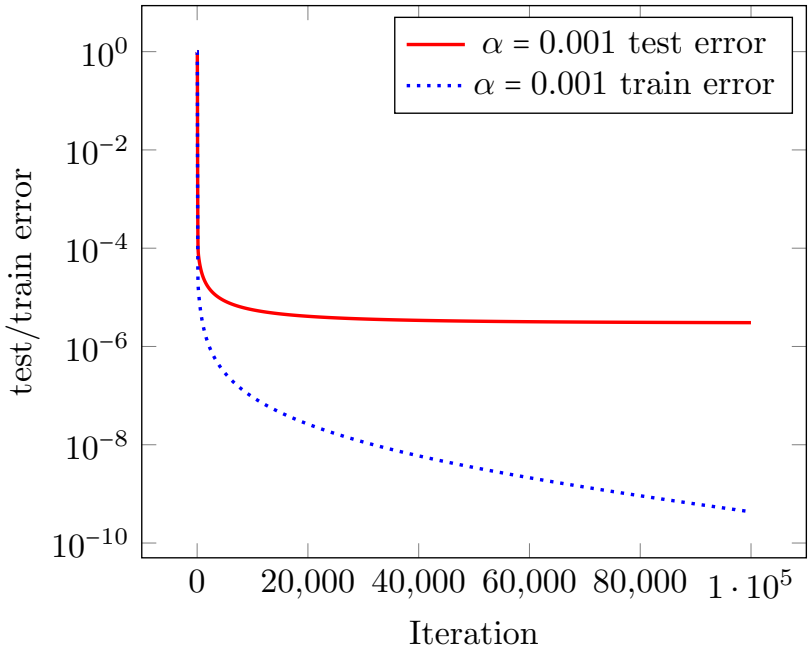
$$\lim_{\alpha \to 0} \lim_{t \to \infty} \boldsymbol{V}_t \boldsymbol{W}_t^\top = \boldsymbol{X}_*$$

Simulations

Parameters: $n_1 = 100,\ n_2 = 50,\ r_\star = 5,\ k = 40,\ m = 2000$



Lazy Training regime

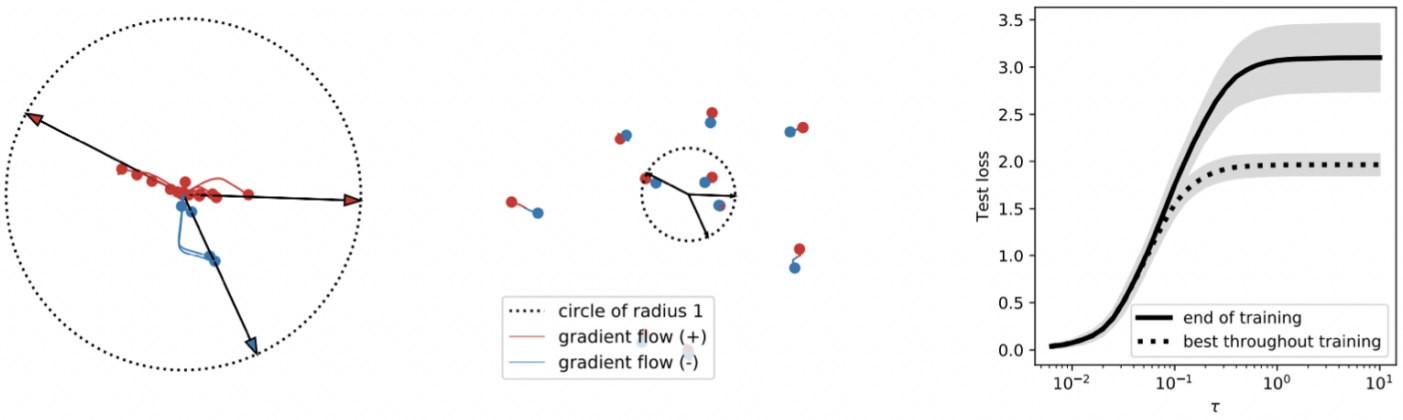


"Feature learning" regime (**this talk!**)

- Test error: $\|\boldsymbol{X}_* - \boldsymbol{V}_t \boldsymbol{W}_t^\top\|_F^2$
- Train error: $\mathcal{L}(\boldsymbol{V}_t, \boldsymbol{W}_t)$

Connection to neural networks?!

Role of scale of initialization in training neural networks



(a) Non-lazy training ($\tau = 0.1$) (b) Lazy training ($\tau = 2$) (c) Generalization properties

Figure 1: Training a two-layer ReLU neural network initialized with normal random weights of variance τ^2: lazy training occurs when τ is large. (a)-(b) Trajectory of weights during gradient descent in 2-D (color shows sign of output layer). (c) Generalization in 100-D: it worsens as τ increases. The ground truth is generated with 3 neurons (arrows in (a)-(b)). Details in Section 3.

[1] Figure from Chizat, Oyallon, and Bach '19: On Lazy Training in Differentiable Programming, NeurIPS '19

Challenge

How to establish generalization and convergence of vanilla gradient descent in low-rank matrix recovery from small random initialization?

Previous work

Certain special cases have been analysed in previous work.

- Case 1: $\boldsymbol{X}_* = \boldsymbol{M}\boldsymbol{M}^\top$ is positive semidefinite (Approach: Minimize $\tilde{\mathcal{L}}(\boldsymbol{U}) = \frac{1}{4m}\sum_{i=1}^{m}\left(y_i - \langle \boldsymbol{A}_i, \boldsymbol{U}\boldsymbol{U}^\top\rangle\right)^2$)

Previous work

Certain special cases have been analysed in previous work.

- Case 1: $\boldsymbol{X}_* = \boldsymbol{M}\boldsymbol{M}^\top$ is positive semidefinite (Approach: Minimize $\tilde{\mathcal{L}}(\boldsymbol{U}) = \frac{1}{4m}\sum_{i=1}^m \left(y_i - \langle \boldsymbol{A}_i, \boldsymbol{U}\boldsymbol{U}^\top \rangle\right)^2$)
  - Case $k = n$ has been solved in (Li, Ma, Zhang COLT '18)
  - Extended to general case $r_\star \leq k \leq n$ in (**DS**, Soltanolkotabi NeurIPS '21) + further improvements

Previous work

Certain special cases have been analysed in previous work.

- Case 1: $\boldsymbol{X}_* = \boldsymbol{M}\boldsymbol{M}^\top$ is positive semidefinite (Approach: Minimize $\tilde{\mathcal{L}}(\boldsymbol{U}) = \frac{1}{4m}\sum_{i=1}^m \left(y_i - \langle \boldsymbol{A}_i, \boldsymbol{U}\boldsymbol{U}^\top \rangle\right)^2$)
  - Case $k = n$ has been solved in (Li, Ma, Zhang COLT '18)
  - Extended to general case $r_\star \leq k \leq n$ in (**DS**, Soltanolkotabi NeurIPS '21) + further improvements
- Case 2: Asymmetric scenario in population loss, i.e., $\mathcal{L}(\boldsymbol{V}, \boldsymbol{W}) = \frac{1}{2}\|\boldsymbol{X}_* - \boldsymbol{V}\boldsymbol{W}^\top\|_F^2$
  - (Ye, Du NeurIPS '21) analysed the case $k = r_\star$
  - Improved analysis for $k \geq r_\star$ in (Jiang, Chen, and Ding; SIAM Data Science '23)

Previous work

Certain special cases have been analysed in previous work.

- Case 1: $\boldsymbol{X}_* = \boldsymbol{M}\boldsymbol{M}^\top$ is positive semidefinite (Approach: Minimize $\tilde{\mathcal{L}}(\boldsymbol{U}) = \frac{1}{4m}\sum_{i=1}^m \left(y_i - \langle \boldsymbol{A}_i, \boldsymbol{U}\boldsymbol{U}^\top\rangle\right)^2$)
  - Case $k = n$ has been solved in (Li, Ma, Zhang COLT '18)
  - Extended to general case $r_\star \leq k \leq n$ in (**DS**, Soltanolkotabi NeurIPS '21) + further improvements
- Case 2: Asymmetric scenario in population loss, i.e., $\mathcal{L}(\boldsymbol{V}, \boldsymbol{W}) = \frac{1}{2}\|\boldsymbol{X}_* - \boldsymbol{V}\boldsymbol{W}^\top\|_F^2$
  - (Ye, Du NeurIPS '21) analysed the case $k = r_\star$
  - Improved analysis for $k \geq r_\star$ in (Jiang, Chen, and Ding; SIAM Data Science '23)
  - **Problem:** No "real" implicit regularization as all global minimizers $(\boldsymbol{V}, \boldsymbol{W})$ satisfy $\boldsymbol{X}_* = \boldsymbol{V}\boldsymbol{W}^\top$

Can we understand the asymmetric scenario with finitely many samples?

Our result

For simplicity, assume $\kappa := \frac{\|\boldsymbol{X}_*\|}{\sigma_{r_\star}(\boldsymbol{X}_*)} \asymp 1$ and Gaussian matrices $\boldsymbol{A}_i$

Our result

For simplicity, assume $\kappa := \frac{\|\boldsymbol{X}_*\|}{\sigma_{r_\star}(\boldsymbol{X}_*)} \asymp 1$ and Gaussian matrices $\boldsymbol{A}_i$

Theorem (Soltanolkotabi, DS, Xie COLT '23)

Assume

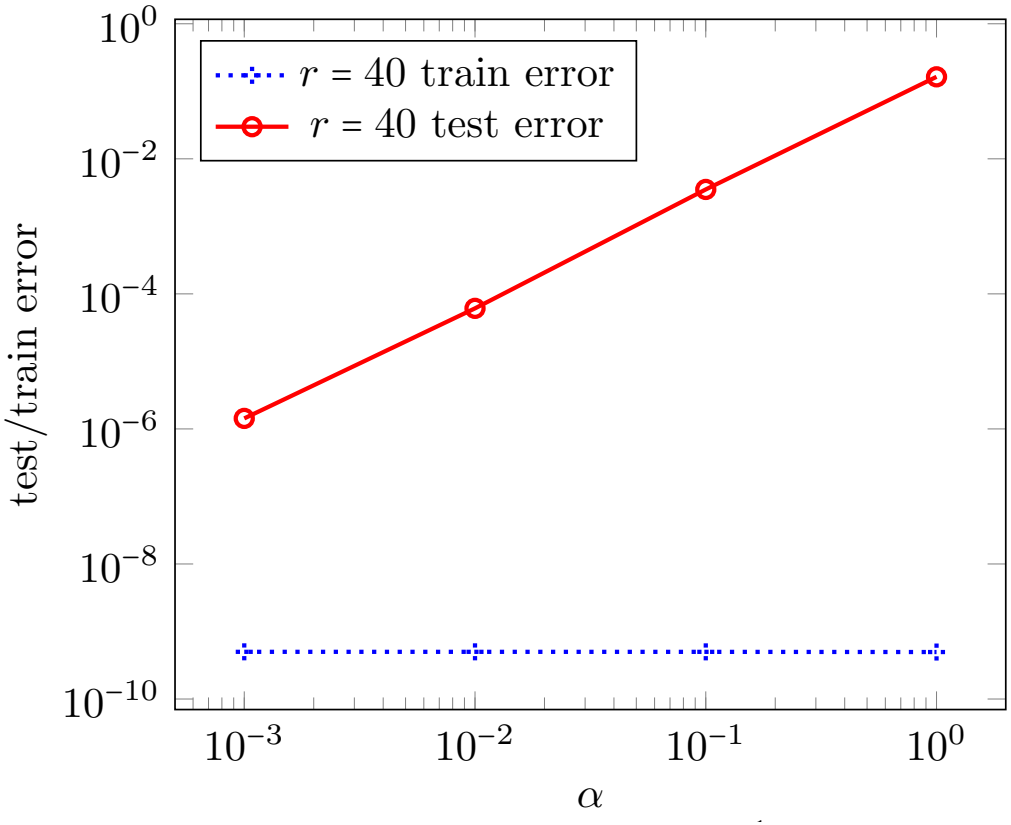
- $m \gtrsim r_\star^2 n$
- $k \geq r_\star$
- small, random initialization
  - $\boldsymbol{V}_0 = \alpha \overline{\boldsymbol{V}}$, $\boldsymbol{W}_0 = \alpha \overline{\boldsymbol{W}}$ with $\overline{\boldsymbol{V}}, \overline{\boldsymbol{W}} \in \mathbb{R}^{n \times k}$ i.i.d. $\mathcal{N}(0,1)$ entries
  - $\alpha \leq \ldots$
- step size $\mu \lesssim \frac{1}{\log(\|\boldsymbol{X}_*\|/\alpha)} \cdot \frac{1}{\|\boldsymbol{X}_*\|}$

Then, w.h.p., $(\boldsymbol{V_T}, \boldsymbol{W_T})$, where $T \asymp \frac{\log(n\|\boldsymbol{X}_*\|/\alpha)}{\mu \sigma_{\min}(\boldsymbol{X}_*)}$, satisfies

$$\frac{\|\boldsymbol{X}_* - \boldsymbol{V_T}\boldsymbol{W_T}^\top\|_F}{\|\boldsymbol{X}_*\|} \lesssim \text{poly}\,(n, r_\star, k)\, \frac{\alpha^{3/5}}{\|\boldsymbol{X}_*\|^{3/10}}$$

Does our theory match practice?

Setting: $n_1 = 100$, $n_2 = 50$, $r_\star = 5$, $k = 40$



- Train error: $\mathcal{L}(\boldsymbol{V_t}, \boldsymbol{W_t})$
- (Relative) Test error: $\frac{\|\boldsymbol{X}_* - \boldsymbol{V_t}\boldsymbol{W_t}^\top\|_F^2}{\|\boldsymbol{X}_*\|_F^2}$

Some comments about the result

- Sample complexity m is optimal up to a factor of $r_\star$

Some comments about the result

- Sample complexity m is optimal up to a factor of $r_\star$
- Result allows for arbitrary amount of overparameterization

Some comments about the result

- Sample complexity m is optimal up to a factor of $r_\star$
- Result allows for arbitrary amount of overparameterization
- Even for $k = r_\star$, this is the first result which shows convergence of non-convex gradient descent from random initialization in the finite sample scenario

A glimpse of the analysis

Proof ideas

Ingredient 1: Symmetrization

Set

$$\boldsymbol{sym}(\boldsymbol{X}_\star) := \begin{bmatrix} 0 & \boldsymbol{X}_* \\ \boldsymbol{X}_*^\top & 0 \end{bmatrix}, \boldsymbol{Z}_t := \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{V}_t \\ \boldsymbol{W}_t \end{bmatrix}, \text{ and } \tilde{\boldsymbol{Z}}_t := \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{V}_t \\ -\boldsymbol{W}_t \end{bmatrix}$$

Proof ideas

Ingredient 1: Symmetrization

Set

$$\boldsymbol{sym}(\boldsymbol{X}_\star) := \begin{bmatrix} 0 & \boldsymbol{X}_* \\ \boldsymbol{X}_*^\top & 0 \end{bmatrix}, \boldsymbol{Z}_t := \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{V}_t \\ \boldsymbol{W}_t \end{bmatrix}, \text{ and } \tilde{\boldsymbol{Z}}_t := \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{V}_t \\ -\boldsymbol{W}_t \end{bmatrix}$$

It follows that

$$\boldsymbol{Z}_t \boldsymbol{Z}_t^\top - \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top = \begin{bmatrix} 0 & \boldsymbol{V}_t \boldsymbol{W}_t^\top \\ \boldsymbol{W}_t \boldsymbol{V}_t^\top & 0 \end{bmatrix}.$$

Define

$$\boldsymbol{B}_i := \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & \boldsymbol{A}_i \\ \boldsymbol{A}_i^\top & 0 \end{bmatrix} \quad \text{for } i \in \{1; \ldots; m\}$$

Proof ideas

Ingredient 1 (continued): Symmetrization

A computation shows

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{V}_t, \boldsymbol{W}_t\right) &= \frac{1}{2m}\sum_{i=1}^{m}\left(\langle \boldsymbol{A}_i, \boldsymbol{X}_* - \boldsymbol{V}_t\boldsymbol{V}_t^\top\rangle\right)^2 \\ &= \frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star) - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2 \\ &=: \mathcal{L}_{\mathsf{Sym}}\left(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t\right)\end{aligned}$$

Proof ideas

Ingredient 1 (continued): Symmetrization

A computation shows

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{V}_t, \boldsymbol{W}_t\right) &= \frac{1}{2m}\sum_{i=1}^{m}\left(\langle \boldsymbol{A}_i, \boldsymbol{X}_* - \boldsymbol{V}_t\boldsymbol{V}_t^\top\rangle\right)^2 \\ &= \frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star) - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2 \\ &=: \mathcal{L}_{\mathsf{Sym}}\left(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t\right)\end{aligned}$$

Instead of $\mathcal{L}$ we equivalently analyze the symmetrized function $\mathcal{L}_{\mathsf{Sym}}$!

Proof ideas

Ingredient 1 (continued): Symmetrization

A computation shows

$$\begin{aligned}\mathcal{L}(\boldsymbol{V}_t, \boldsymbol{W}_t) &= \frac{1}{2m}\sum_{i=1}^{m}\left(\langle \boldsymbol{A}_i, \boldsymbol{X}_* - \boldsymbol{V}_t\boldsymbol{V}_t^\top\rangle\right)^2 \\ &= \frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star) - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2 \\ &=: \mathcal{L}_{\mathrm{Sym}}\left(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t\right)\end{aligned}$$

Instead of $\mathcal{L}$ we equivalently analyze the symmetrized function $\mathcal{L}_{\mathrm{Sym}}$!

In other words, we analyze trajectories of $(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t)$ instead of $(\boldsymbol{V}_t, \boldsymbol{V}_t)$
$\Longrightarrow$ Allows us to use proof techniques which were established for the Positive Semidefinite (PSD) case

Proof ideas

Ingredient 2: Decomposition into signal and nuisance term

- Decompose $\boldsymbol{sym}(\boldsymbol{X}_\star)$ into its positive definite and negative definite part

$$\boldsymbol{sym}(\boldsymbol{X}_\star) = \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{sym}(\boldsymbol{X}_\star)^-$$

Proof ideas

Ingredient 2: Decomposition into signal and nuisance term

- Decompose $\boldsymbol{sym}(\boldsymbol{X}_\star)$ into its positive definite and negative definite part

$$\boldsymbol{sym}(\boldsymbol{X}_\star) = \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{sym}(\boldsymbol{X}_\star)^-$$

- Consider eigendecomposition $\boldsymbol{sym}(\boldsymbol{X}_\star)^+ = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ with $\boldsymbol{V} \in \mathbb{R}^{n \times r_\star}$

Proof ideas

Ingredient 2: Decomposition into signal and nuisance term

- Decompose $\boldsymbol{sym}(\boldsymbol{X}_\star)$ into its positive definite and negative definite part

$$\boldsymbol{sym}(\boldsymbol{X}_\star) = \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{sym}(\boldsymbol{X}_\star)^-$$

- Consider eigendecomposition $\boldsymbol{sym}(\boldsymbol{X}_\star)^+ = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ with $\boldsymbol{V} \in \mathbb{R}^{n \times r_\star}$
- Let $\boldsymbol{P}_t \boldsymbol{\Sigma}_t \boldsymbol{Q}_t$ be the singular value decomposition of $\boldsymbol{V}^\top \boldsymbol{Z}_t$. Decompose

$$\boldsymbol{Z}_t = \underbrace{\boldsymbol{Z}_t \boldsymbol{Q}_t \boldsymbol{Q}_t^\top}_{\text{signal term}} + \underbrace{\boldsymbol{Z}_t \left(\boldsymbol{Id} - \boldsymbol{Q}_t \boldsymbol{Q}_t^\top \right)}_{\text{nuisance term}}$$

We show that $\boldsymbol{Z}_t \boldsymbol{Q}_t \boldsymbol{Q}_t^\top \boldsymbol{Z}_t^\top$ converges towards $\boldsymbol{sym}(\boldsymbol{X}_\star)^+$, whereas the nuisance term stays relatively small

Proof ideas

Ingredient 2: Decomposition into signal and nuisance term

- Decompose $\boldsymbol{sym}(\boldsymbol{X}_\star)$ into its positive definite and negative definite part

$$\boldsymbol{sym}(\boldsymbol{X}_\star) = \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{sym}(\boldsymbol{X}_\star)^-$$

- Consider eigendecomposition $\boldsymbol{sym}(\boldsymbol{X}_\star)^+ = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ with $\boldsymbol{V} \in \mathbb{R}^{n \times r_\star}$
- Let $\boldsymbol{P}_t \boldsymbol{\Sigma}_t \boldsymbol{Q}_t$ be the singular value decomposition of $\boldsymbol{V}^\top \boldsymbol{Z}_t$. Decompose

$$\boldsymbol{Z}_t = \underbrace{\boldsymbol{Z}_t \boldsymbol{Q}_t \boldsymbol{Q}_t^\top}_{\text{signal term}} + \underbrace{\boldsymbol{Z}_t \left(\boldsymbol{Id} - \boldsymbol{Q}_t \boldsymbol{Q}_t^\top \right)}_{\text{nuisance term}}$$

We show that $\boldsymbol{Z}_t \boldsymbol{Q}_t \boldsymbol{Q}_t^\top \boldsymbol{Z}_t^\top$ converges towards $\boldsymbol{sym}(\boldsymbol{X}_\star)^+$, whereas the nuisance term stays relatively small

(Similar decomposition for $\tilde{\boldsymbol{Z}}_t$.)

Proof ideas

Ingredient 3: Decoupling

- **Main challenge compared to PSD case:** Two factors $\boldsymbol{Z}_t$ and $\tilde{\boldsymbol{Z}}_t$ whose trajectories are coupled with each other
- How to deal with this coupling?

Proof ideas

Ingredient 3: Decoupling

- **Main challenge compared to PSD case:** Two factors $\boldsymbol{Z}_t$ and $\tilde{\boldsymbol{Z}}_t$ whose trajectories are coupled with each other
- How to deal with this coupling? We observe

$$\begin{aligned}\boldsymbol{Z}_{t+1} =& \boldsymbol{Z}_t - \mu \nabla_{\boldsymbol{Z}} \mathcal{L}(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t) \\ =& \boldsymbol{Z}_t + \mu \frac{1}{m} \left(\sum_{i=1}^{m} \langle \boldsymbol{A}_i, \boldsymbol{X}_* - \boldsymbol{Z}_t \boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top \rangle \boldsymbol{A}_i \right) \boldsymbol{Z}_t \\ =& \boldsymbol{Z}_t + \mu \cdot \underbrace{\frac{1}{m} \left(\boldsymbol{X}_* - \boldsymbol{Z}_t \boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top \right) \boldsymbol{Z}_t}_{\text{population gradient}} + \mu \boldsymbol{\Delta}_t\end{aligned}$$

Proof ideas

Ingredient 3: Decoupling

- **Main challenge compared to PSD case:** Two factors $\boldsymbol{Z}_t$ and $\tilde{\boldsymbol{Z}}_t$ whose trajectories are coupled with each other
- How to deal with this coupling? We observe

$$\begin{aligned}\boldsymbol{Z}_{t+1} =& \boldsymbol{Z}_t - \mu \nabla_{\boldsymbol{Z}} \mathcal{L}(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t) \\ =& \boldsymbol{Z}_t + \mu \frac{1}{m} \left(\sum_{i=1}^{m} \langle \boldsymbol{A}_i, \boldsymbol{X}_* - \boldsymbol{Z}_t \boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top \rangle \boldsymbol{A}_i \right) \boldsymbol{Z}_t \\ =& \boldsymbol{Z}_t + \mu \cdot \underbrace{\frac{1}{m} \left(\boldsymbol{X}_* - \boldsymbol{Z}_t \boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top \right) \boldsymbol{Z}_t}_{\text{population gradient}} + \mu \boldsymbol{\Delta}_t\end{aligned}$$

where

$$\boldsymbol{\Delta}_t := \left(\boldsymbol{X}_* - \boldsymbol{Z}_t \boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top - \frac{1}{m} \sum_{i=1}^{m} \langle \boldsymbol{A}_i, \boldsymbol{X}_* - \boldsymbol{Z}_t \boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t \tilde{\boldsymbol{Z}}_t^\top \rangle \boldsymbol{A}_i \right) \boldsymbol{Z}_t$$

can be interpreted as some sort of noise term.

Proof ideas

- We computed that

$$\boldsymbol{Z}_{t+1} = \boldsymbol{Z}_t + \frac{\mu}{m}\left(\boldsymbol{X}_* - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\right)\boldsymbol{Z}_t + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + \boldsymbol{\Delta}_t$$

- Coupling is determined by the **imbalance term**

$$\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t = \frac{1}{2}\left(\boldsymbol{V}_t^\top\boldsymbol{V}_t - \boldsymbol{W}_t^\top\boldsymbol{W}_t\right).$$

Proof ideas

- We computed that

$$\boldsymbol{Z}_{t+1} = \boldsymbol{Z}_t + \frac{\mu}{m}\left(\boldsymbol{X}_* - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\right)\boldsymbol{Z}_t + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + \boldsymbol{\Delta}_t$$

- Coupling is determined by the **imbalance term**

$$\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t = \frac{1}{2}\left(\boldsymbol{V}_t^\top\boldsymbol{V}_t - \boldsymbol{W}_t^\top\boldsymbol{W}_t\right).$$

- Direct (and well-known) calculation shows

$$\tilde{\boldsymbol{Z}}_{t+1}^\top\boldsymbol{Z}_{t+1} = \tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + O(\mu^2)$$

Proof ideas

- We computed that

$$\boldsymbol{Z}_{t+1} = \boldsymbol{Z}_t + \frac{\mu}{m}\left(\boldsymbol{X}_* - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\right)\boldsymbol{Z}_t + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + \boldsymbol{\Delta}_t$$

- Coupling is determined by the **imbalance term**

$$\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t = \frac{1}{2}\left(\boldsymbol{V}_t^\top\boldsymbol{V}_t - \boldsymbol{W}_t^\top\boldsymbol{W}_t\right).$$

- Direct (and well-known) calculation shows

$$\tilde{\boldsymbol{Z}}_{t+1}^\top\boldsymbol{Z}_{t+1} = \tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + O(\mu^2)$$

- Moreover,

$$\tilde{\boldsymbol{Z}}_0^\top\boldsymbol{Z}_0 = O(\alpha^2)$$

Proof ideas

- We computed that

$$\boldsymbol{Z}_{t+1} = \boldsymbol{Z}_t + \frac{\mu}{m}\left(\boldsymbol{X}_* - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\right)\boldsymbol{Z}_t + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + \boldsymbol{\Delta}_t$$

- Coupling is determined by the **imbalance term**

$$\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t = \frac{1}{2}\left(\boldsymbol{V}_t^\top\boldsymbol{V}_t - \boldsymbol{W}_t^\top\boldsymbol{W}_t\right).$$

- Direct (and well-known) calculation shows

$$\tilde{\boldsymbol{Z}}_{t+1}^\top\boldsymbol{Z}_{t+1} = \tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t + O(\mu^2)$$

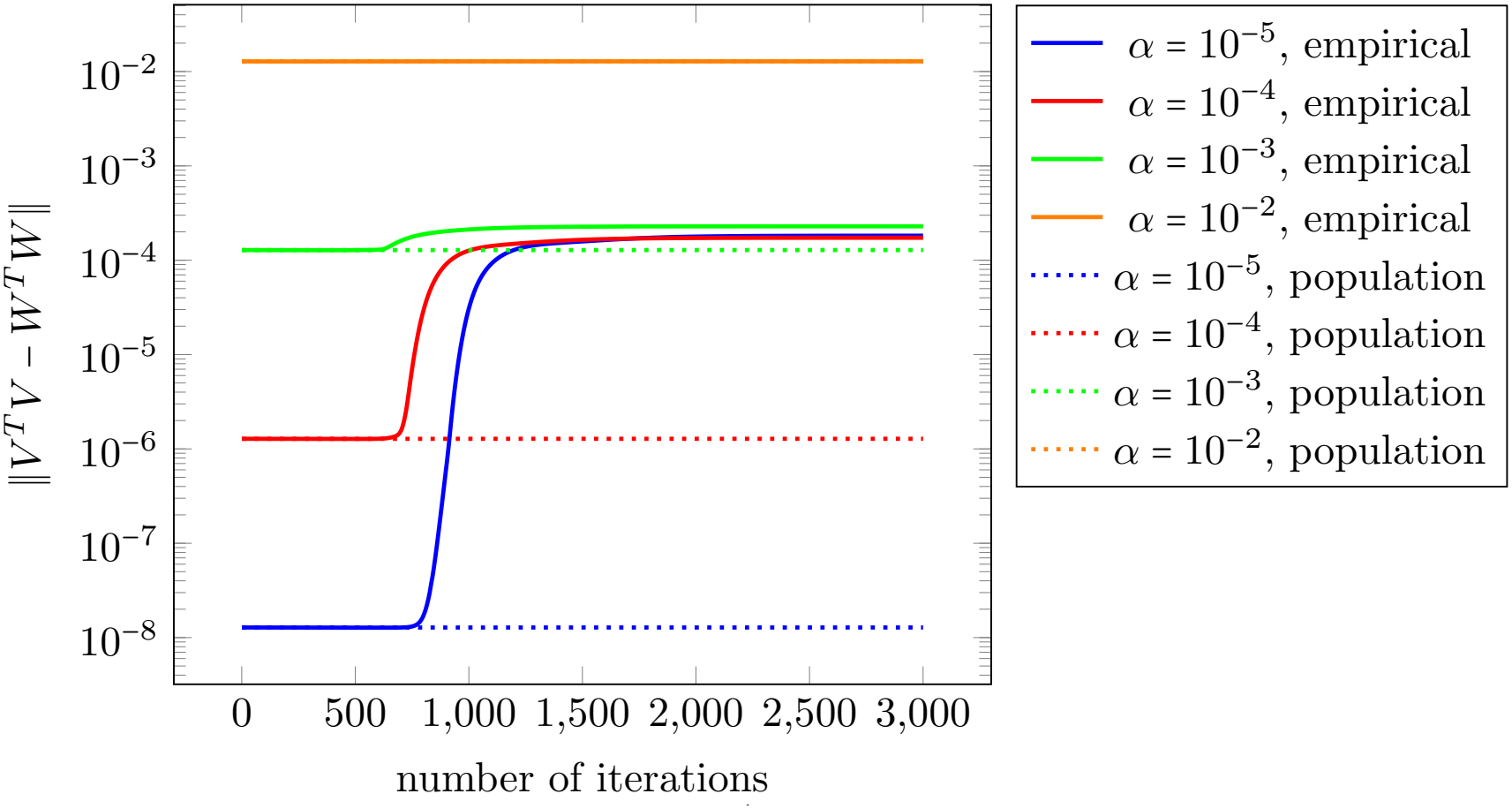
- Moreover,

$$\tilde{\boldsymbol{Z}}_0^\top\boldsymbol{Z}_0 = O(\alpha^2)$$

- **(Naive) Approach:** For small initialization and sufficiently small step size, $\|\tilde{\boldsymbol{Z}}_t^\top\boldsymbol{Z}_t\|$ stays small, leading to decoupled trajectories

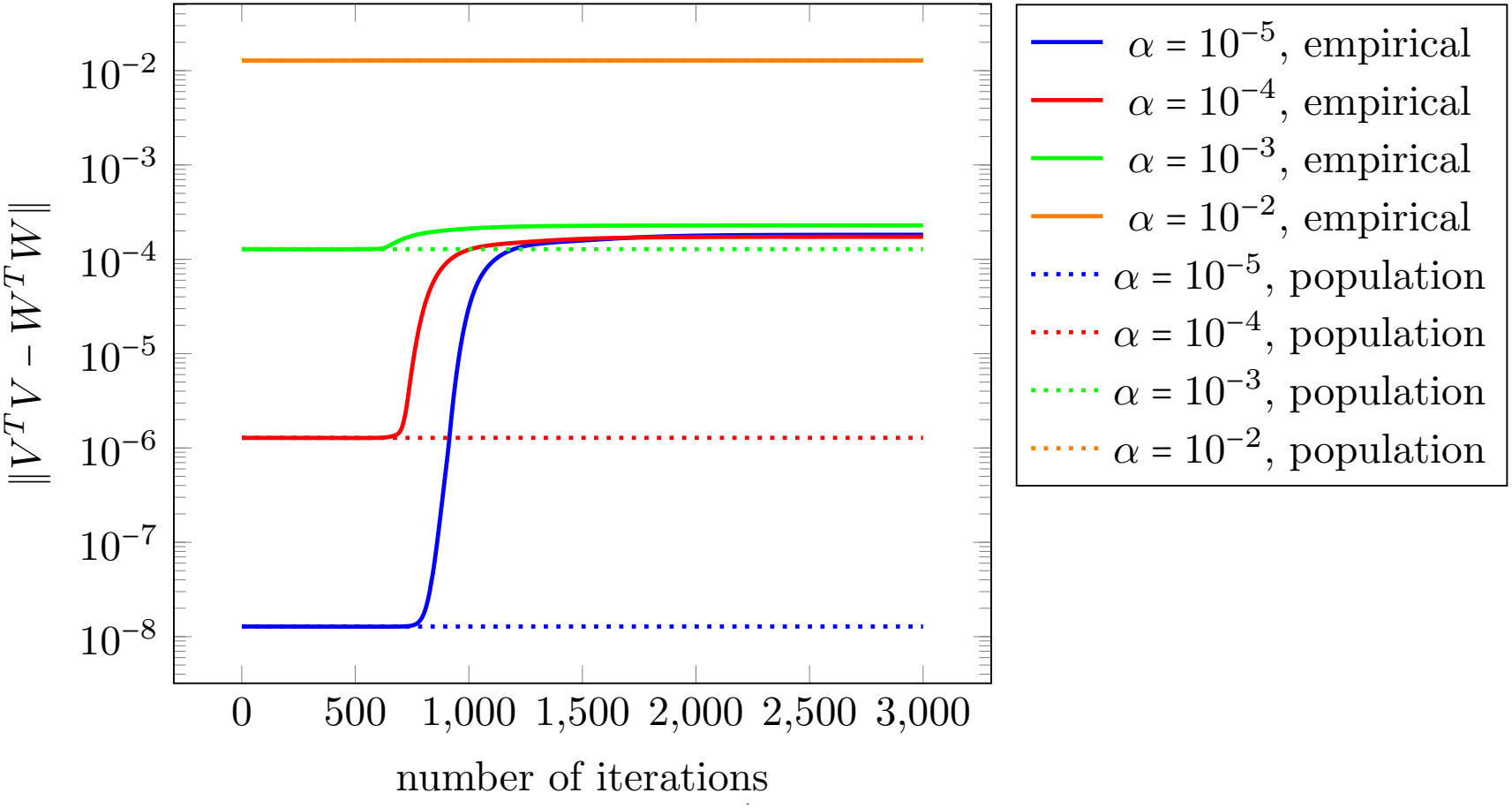
What happens for realistic step sizes?

Parameters: $n_1 = 100$, $n_2 = 50$, $r_\star = 5$, $k = 40$, $m = 2000$, $\mu = \frac{1}{100\|X\|}$



What happens for realistic step sizes?

Parameters: $n_1 = 100$, $n_2 = 50$, $r_\star = 5$, $k = 40$, $m = 2000$, $\mu = \frac{1}{100\|X\|}$



For realistic step sizes μ, the imbalance term grows significantly! We cannot just assume that the imbalance term stays small!

$\Rightarrow$ Refined analysis is needed!

Proof ideas

Ingredient 3: Decoupling (continued)

- In addition to $\|\tilde{\boldsymbol{Z}}_t^\top \boldsymbol{Z}_t\|$ analyze the quantities

$$\|\tilde{\boldsymbol{Z}}_t^\top \boldsymbol{Z}_t \left(\mathrm{Id} - \boldsymbol{Q}_t \boldsymbol{Q}_t^\top\right)\| \quad \text{(imbalance term parallel to nuisance part)}$$

$$\|\tilde{\boldsymbol{Z}}_t^\top \boldsymbol{P}_{Z_t Q_t}\| \quad \text{(imbalance term parallel to the signal part)}$$

where $\boldsymbol{P}_{Z_t Q_t} \in \mathbb{R}^{n \times r_\star}$ is the matrix which contains the left-singular vectors of $\boldsymbol{Z}_t \boldsymbol{Q}_t$

Ingredient 3: Decoupling (continued)

- In addition to $\|\tilde{\boldsymbol{Z}}_t^\top \boldsymbol{Z}_t\|$ analyze the quantities

$$\|\tilde{\boldsymbol{Z}}_t^\top \boldsymbol{Z}_t \left(\mathrm{Id} - \boldsymbol{Q}_t \boldsymbol{Q}_t^\top\right)\| \quad \text{(imbalance term parallel to nuisance part)}$$

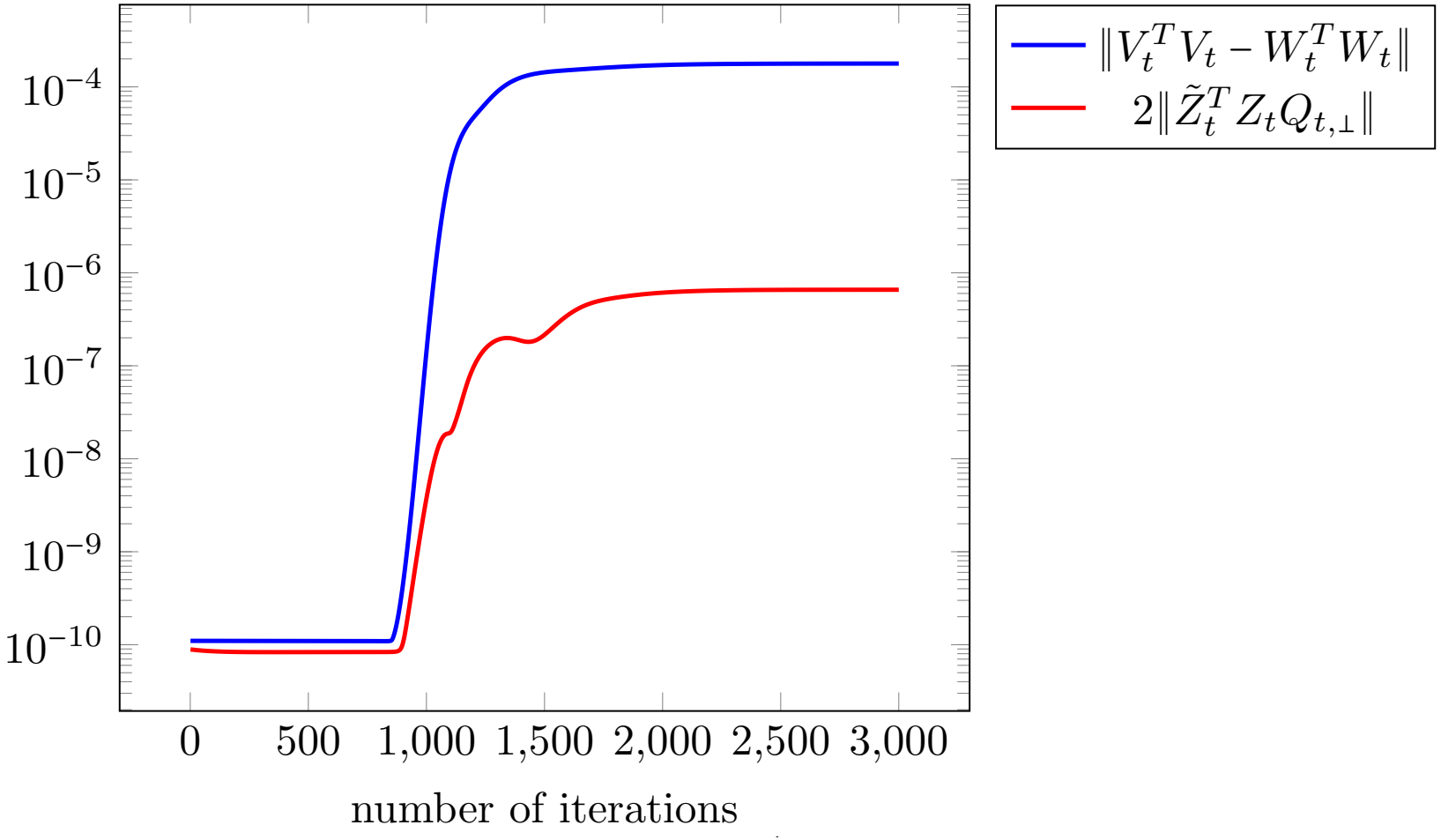
$$\|\tilde{\boldsymbol{Z}}_t^\top \boldsymbol{P}_{Z_t Q_t}\| \quad \text{(imbalance term parallel to the signal part)}$$

  where $\boldsymbol{P}_{Z_t Q_t} \in \mathbb{R}^{n \times r_\star}$ is the matrix which contains the left-singular vectors of $\boldsymbol{Z}_t \boldsymbol{Q}_t$

- Showing that these quantities stay small we can show that the trajectories of $\boldsymbol{Z}_t$ and $\tilde{\boldsymbol{Z}}_t$ are (almost) decoupled!

Simulations

Parameters: $n_1 = 100$, $n_2 = 50$, $r_\star = 5$, $k = 10$, $m = 2000$, $\mu = \frac{1}{100\|X\|}$



Proof ideas

Ingredient 3: Decoupling (continued)

Morally speaking, by showing that the imbalance terms stay small we obtain that

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t\right) =& \frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star) - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2 \\ \approx& \underbrace{\frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\rangle\right)^2}_{=:\mathcal{L}_1(\boldsymbol{Z}_t)} \\ &+ \underbrace{\frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star)^- - \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2}_{=:\mathcal{L}_2(\tilde{\boldsymbol{Z}}_t)}\end{aligned}$$

Proof ideas

Ingredient 3: Decoupling (continued)

Morally speaking, by showing that the imbalance terms stay small we obtain that

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t\right) =& \frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star) - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2 \\ \approx& \underbrace{\frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\rangle\right)^2}_{=:\mathcal{L}_1(\boldsymbol{Z}_t)} \\ &+ \underbrace{\frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star)^- - \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2}_{=:\mathcal{L}_2(\tilde{\boldsymbol{Z}}_t)}\end{aligned}$$

Hence, it suffices to understand the dynamics of $\boldsymbol{Z}_t$ and $\tilde{\boldsymbol{Z}}_t$ individually!

Proof ideas

Ingredient 3: Decoupling (continued)

Morally speaking, by showing that the imbalance terms stay small we obtain that

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{Z}_t, \tilde{\boldsymbol{Z}}_t\right) =& \frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star) - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top + \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2 \\ \approx& \underbrace{\frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star)^+ - \boldsymbol{Z}_t\boldsymbol{Z}_t^\top\rangle\right)^2}_{=:\mathcal{L}_1(\boldsymbol{Z}_t)} \\ &+ \underbrace{\frac{1}{4m}\sum_{i=1}^{m}\left(\langle \boldsymbol{B}_i, \boldsymbol{sym}(\boldsymbol{X}_\star)^- - \tilde{\boldsymbol{Z}}_t\tilde{\boldsymbol{Z}}_t^\top\rangle\right)^2}_{=:\mathcal{L}_2(\tilde{\boldsymbol{Z}}_t)}\end{aligned}$$

Hence, it suffices to understand the dynamics of $\boldsymbol{Z}_t$ and $\tilde{\boldsymbol{Z}}_t$ individually!$\Rightarrow$ These can be analyzed using techniques from DS, Soltanolkotabi '21!

Conclusion

We show that in overparameterized matrix sensing gradient descent with **small, random initialization** converges to the low-rank solution.

Conclusion

We show that in overparameterized matrix sensing gradient descent with **small, random initialization** converges to the low-rank solution.

Lots of open questions:

- More structured measurements (such as matrix completion)?!
- Can we remove the additional $r_\star$-factor in the sample complexity?
- Can we remove the early stopping assumptions in our main result?
- What about other mechanism that induce implicit bias towards low-rank solutions such as large step sizes with label noise?!
- Overparameterization and small, random initialization in neural networks?
- ...

References

1. D. Stöger and M. Soltanolkotabi:
   Small random initialization is akin to spectral learning: Optimization and generalization guarantees for overparameterized low-rank matrix reconstruction.
   NeurIPS 2021
2. M. Soltanolkotabi, D. Stöger, C. Xie:
   Implicit Balancing and Regularization: Generalization and Convergence Guarantees for Overparameterized Asymmetric Matrix Sensing.
   COLT 2023