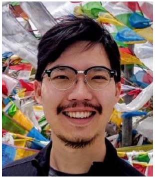
Feature learning in two-layer neural networks under structured data

Denny Wu

Center for Data Science, New York University
Center for Computational Mathematics, Flatiron Institute

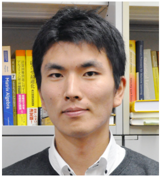
Introduction

- "*High-dimensional asymptotics of feature learning: how one gradient step improves the representation*", **NeurIPS 2022 (short version)**.
- "*Learning in the presence of low-dimensional structure: a spiked random matrix perspective*", **NeurIPS 2023**.
- "*Gradient-based feature learning under structured data*", **NeurIPS 2023**.


Jimmy Ba


Murat A. Erdogdu


Alireza Mousavi


Taiji Suzuki


Zhichao Wang


Greg Yang

Introduction: Learning under Structured Data

Target function: *low-dimensional* polynomial $f_* : \mathbb{R}^d \to \mathbb{R}$

Single-index target (teacher)[1]: $f_*(\boldsymbol{x}) = \sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}_* \rangle), \ \boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$.

- **Link function** $\sigma_* : \mathbb{R} \to \mathbb{R}$ is a degree-p polynomial (with $\mathbb{E}_{\mathcal{N}(0,1)}[\sigma_*] = 0$).

Input Data: *high-dimensional* feature with *low-dimensional* structure

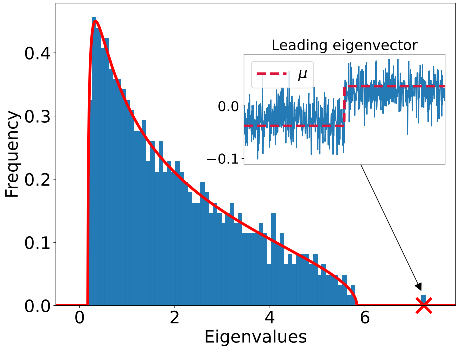
Spiked covariance data: $\boldsymbol{\Sigma} = \boldsymbol{I} + \theta \boldsymbol{\mu}\boldsymbol{\mu}^\top, \quad \|\boldsymbol{\mu}\| = 1, \ \theta \asymp d^\beta$.

- **High-dimensionality:** large amount of input features ($d \to \infty$).
- **Low-dimensional structure:** Larger spike $\theta \Rightarrow$ stronger anisotropy.

[1] $\boldsymbol{\beta}_*$ is normalized such that $\mathbb{E}\langle \boldsymbol{x}, \boldsymbol{\beta}_* \rangle^2 = 1$, and σ_* is dimension-free.

Introduction: Spiked Random Matrix Model

Spiked Random Matrix: low-dimensional signal + high-dimensional noise.



- **Bulk:** *uninformative* & high-dimensional random noise.
- **Spike:** *informative* & low-dimensional structure.

- Johnstone 2001. On the distribution of the largest eigenvalue in principal components analysis
- Baik et al. 2005. Phase transition of the largest eigenvalue for non-null complex sample covariance matrices

Introduction: Summary of Results

- **Training.** Empirical risk minimization (potentially ℓ_2-regularized):

$$\mathcal{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n}(f(\boldsymbol{x}_i) - y_i)^2, \quad y_i = f_*(\boldsymbol{x}_i) + \varepsilon_i,$$

- **Test.** Prediction risk: $\mathcal{R}(f) = \mathbb{E}_{\boldsymbol{x}}[(f(\boldsymbol{x}) - f_*(\boldsymbol{x}))^2] = \|f - f_*\|^2_{L^2(P_x)}$.

Overview: complexity of gradient-based feature learning

❐ interplay between structured data and statistical & optimization efficiency.

---

1. **one-step feature learning**: sharp guarantees in the *proportional regime*.
2. **(normalized) gradient flow** for partially aligned data.
3. **mean-field neural networks** for (anisotropic) k-parity classification.

Student Model I: Kernel Ridge Regression

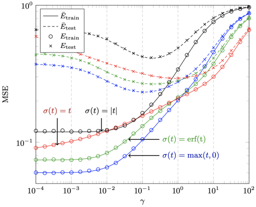
- **Random features regression.** Given $\phi_{\mathrm{RF}}(\boldsymbol{x}) = \frac{1}{\sqrt{N}}\sigma(\boldsymbol{W}_0^\top \boldsymbol{x}) \in \mathbb{R}^N$,

$$\hat{f}_{\mathrm{RF}}(\boldsymbol{x}) = \langle \phi_{\mathrm{RF}}(\boldsymbol{x}), \hat{\boldsymbol{a}} \rangle, \quad \hat{\boldsymbol{a}} = \operatorname{argmin}_{\boldsymbol{a}} \left\{ \frac{1}{n} \sum_{i=1}^n (y_i - \langle \phi_{\mathrm{RF}}(\boldsymbol{x}_i), \boldsymbol{a} \rangle)^2 + \frac{\lambda}{N} \|\boldsymbol{a}\|^2 \right\}.$$

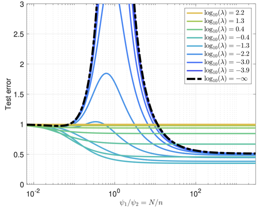
- **Kernel ridge regression.** Given inner-product kernel: $k(\boldsymbol{x}, \boldsymbol{y}) = g\left(\frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{d}\right)$,

$$\hat{f}_{\mathrm{ker}} = \operatorname*{argmin}_{f \in \mathcal{H}} \left\{ \frac{1}{n} \sum_{i=1}^n (y_i - f(\boldsymbol{x}_i))^2 + \lambda \|f\|_{\mathcal{H}}^2 \right\} \Rightarrow \hat{f}_{\mathrm{ker}}(\boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{X})^\top (\boldsymbol{K} + \lambda \boldsymbol{I})^{-1} \boldsymbol{y}.$$

Fixed feature map $\Longrightarrow$ no representation learning.



[Louart, Liao, and Couillet, 2018].



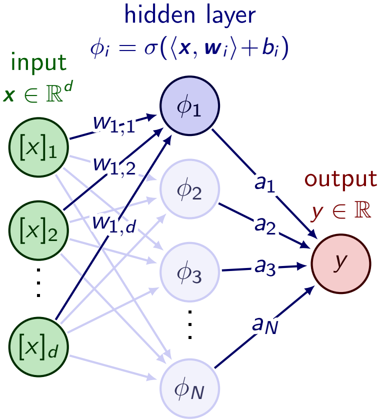
[Mei and Montanari, 2019].

Student Model II: Two-layer Neural Network

Width-N Two-layer NN

$$f_{\text{NN}}(\boldsymbol{x}) = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} a_i \sigma(\langle \boldsymbol{x}, \boldsymbol{w}_i \rangle + b_i)$$

- Input data: $\boldsymbol{x} \in \mathbb{R}^d$.
- Parameters: $\boldsymbol{W} \in \mathbb{R}^{d\times N}$, $\boldsymbol{a} \in \mathbb{R}^N$, $\boldsymbol{b} \in \mathbb{R}^N$.
- Element-wise nonlinearity: $\sigma : \mathbb{R} \to \mathbb{R}$.

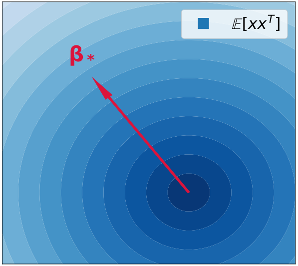


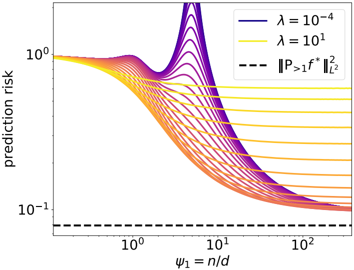
Optimization: given a convex loss ℓ,

- Optimizing $\boldsymbol{a}$ under fixed $\boldsymbol{W}$ is *convex*.
- Optimizing $\boldsymbol{W}$ under fixed $\boldsymbol{a}$ is *non-convex*.

Parameters $\boldsymbol{W}$ learned via *gradient descent* $\Longrightarrow$ representation learning.

Prior Results: Isotropic Data ($\underline{\theta = 0}$)





Theorem ([Ghorbani et al. 19], [Hu and Lu 20], [Bartlett et al. 21], ...)

Denote $P_{>1}$ as the projector orthogonal to constant and linear functions in $L^2(P_X)$, $f(x) = \mu_0 + \mu_1\langle x, \beta_* \rangle + P_{>1}f(x)$. *Then for* $x \sim \mathcal{N}(0, I)$ *and $n, d \to \infty$, $n/d \to \psi$,*

$$\min\{\mathcal{R}_{\mathrm{RF}}(\lambda), \mathcal{R}_{\mathrm{ker}}(\lambda)\} \geq \|P_{>1}f_*\|_{L^2}^2 + o_{d,\mathbb{P}}(1),$$

- In the proportional limit, kernel models can only learn linear functions.

Prior Results: Isotropic Data ($\theta = 0$)

Theorem ([BES+22], [Bietti et al. 22], [Mousavi et al. 22], [Berthier et al. 23]...)

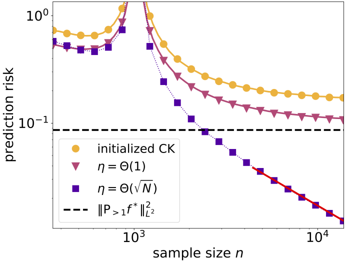
For $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I})$, if the nonlinearities σ, σ_* satisfy a non-degeneracy condition:

$$\mathbb{E}[\sigma'(z)] = \mu_1 \neq 0, \quad \mathbb{E}[\sigma_*'(z)] = \mu_1^* \neq 0, \quad \text{for } z \sim \mathcal{N}(0,1),$$

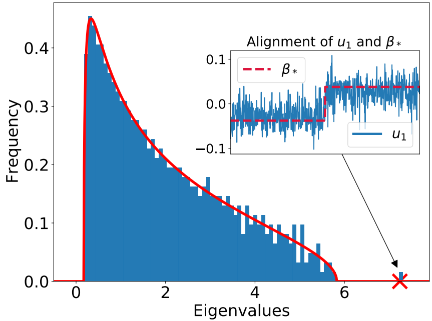
then GD-trained two-layer NN can learn f_* in the proportional regime.

Provable benefit of gradient-based feature learning!

- Small lr $\eta = \Theta(1)$: trained kernel **always improves** upon the initial RF estimator, but the model remains "*linear*".
- Large lr $\eta = \Theta(\sqrt{N})$: regression on trained features can learn **nonlinear** f_*.



A Spiked Model for the Weight Matrix



Blue: empirical simulation

Red: analytic prediction
(*BBP Phase Transition*)

- $\sigma = \tanh$, $f_*(\boldsymbol{x}) = \mathrm{ReLU}(\langle \boldsymbol{x}, \boldsymbol{\beta}_* \rangle)$.
- Teacher $\boldsymbol{\beta}_* \propto [-1_{d/2}; 1_{d/2}]$.

Observation: after *one feature learning step* on the first-layer $\boldsymbol{W}$:

- The **bulk** of the spectrum of $\boldsymbol{W}_1$ remains unchanged
- A **spike** ($\times$) appears in $\boldsymbol{W}_1$, which aligns with signal $\boldsymbol{\beta}^*$

Limitation under Isotropic Data

Question: what if the *nondegeneracy* assumption is violated, i.e. $\mathbb{E}[\sigma_*'(z)] = 0$?

Hermite expansion: $\sigma(z) = \sum_{i=0}^{\infty} \alpha_i \mathrm{He}_i(z)$, $\sigma_*(z) = \sum_{i=0}^{\infty} \alpha_i^* \mathrm{He}_i(z)$.

- we assume $\alpha_0^* = \mathbb{E}[\sigma_*(z)] = 0$.
- **nondegeneracy** $\Rightarrow \alpha_1, \alpha_1^* \neq 0$.

$$
\begin{aligned}
\mathbb{E}[\nabla_{\boldsymbol{w}} \mathcal{L}(f_{\mathrm{NN}})] &\approx \mathbb{E}[\boldsymbol{x}\sigma'(\langle \boldsymbol{x}, \boldsymbol{w}\rangle) f_*(\boldsymbol{x})] \\
&= \boldsymbol{\beta}_* \cdot \mathbb{E}[\sigma_*'(\langle \boldsymbol{x}, \boldsymbol{\beta}_*\rangle)\sigma'(\langle \boldsymbol{x}, \boldsymbol{w}\rangle)] + \boldsymbol{w} \cdot \mathbb{E}[\ldots] && \textit{Stein's lemma} \\
&= \boldsymbol{\beta}_* \cdot \sum_{i=0}^{\infty} (i+1)^2 \alpha_{i+1}\alpha_{i+1}^* \langle \boldsymbol{w}, \boldsymbol{\beta}_*\rangle^i + \ldots && \textit{Hermite expansion}
\end{aligned}
$$

Observation: at random initialization, $\langle \boldsymbol{w}, \boldsymbol{\beta}_*\rangle^i = \tilde{\Theta}(d^{-i/2})$ w.h.p.

Information exponent of σ_*: smallest $k \in \mathbb{N}$ such that $\alpha_k^* \neq 0$.

Intuition: the magnitude of "information" contained in the gradient update.

Limitation under Isotropic Data (continued)

Examples • $\sigma_*(z) = \mathrm{He}_1(z) \Rightarrow k = 1$. • $\sigma_*(z) = \mathrm{He}_3(z) \Rightarrow k = 3$.

• $\sigma_*(z) = \mathrm{He}_1(z) + \mathrm{He}_3(z) \Rightarrow k = 1$. • $\sigma_*(z) = \mathrm{He}_2(z) + \mathrm{He}_3(z) \Rightarrow k = 2$.

Consequence:

- **Gradient norm.** $\|\mathbb{E}[\boldsymbol{x}\boldsymbol{\sigma}'(\langle \boldsymbol{x}, \boldsymbol{w}\rangle) f_*(\boldsymbol{x})]\| = \tilde{\Theta}(d^{-(k-1)/2})$.
- **Gradient concentration.** with high probability,

$$\left\|\mathbb{E}[\boldsymbol{x}\boldsymbol{\sigma}'(\langle \boldsymbol{x}, \boldsymbol{w}\rangle) f_*(\boldsymbol{x})] - \tfrac{1}{n}\textstyle\sum_{i=1}^n \boldsymbol{x}_i \boldsymbol{\sigma}'(\langle \boldsymbol{x}_i, \boldsymbol{w}\rangle) f_*(\boldsymbol{x}_i)\right\| \lesssim \sqrt{d/n}.$$

☹ $n = \Omega(d^k)$ samples required to achieve nontrivial concentration...

In the *proportional regime* $(n \asymp d)$,

- ❒ kernel method only learns linear σ_* (degree $p = 1$).
- ❒ representation learning (with one GD step) only works when $k = 1$.

Motivation: Stronger Learnability Results?

Question: Under what settings can

- kernel ridge regression learn f_* that is *nonlinear* ($p > 1$)?
- two-layer NN + GD learn f_* with *larger information exponent* ($k > 1$)?

Prior results. For *isotropic* $\boldsymbol{x}$, KRR: $n = \Omega(d^p)$, NN: $n = \Omega\left(d^{\Theta(k)}\right)$.

What about the proportional scaling ($n \asymp d$)? Need to introduce *anisotropy*!

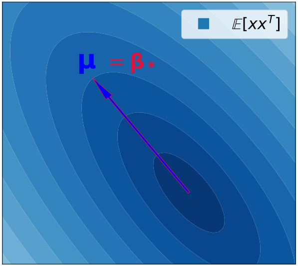
Motivation: if the input already contains low-dimensional structure (spike), can kernel & NN learn a larger class of f_* in the *proportional regime*?

- Ghorbani et al., 2021. Linearized two-layer neural networks in high dimensions
- Ben Arous et al., 2021. Stochastic gradient descent on non-convex losses from high-dimensional inference

Setting: Anisotropic Data with Perfect Alignment

Ideal Setting: perfect alignment between spike and index features $\boldsymbol{\mu} = \boldsymbol{\beta}_*$.

- Can be efficiently solved by PCA + fitting f_* on the top principal component.



- **Spiked data:** $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I} + \theta\boldsymbol{\beta}_*\boldsymbol{\beta}_*^\top)$.
- **Aligned teacher:** $f_*(\boldsymbol{x}) = \sigma_*\left(\frac{1}{\sqrt{1+\theta}}\langle \boldsymbol{x}, \boldsymbol{\beta}_*\rangle\right)$, with degree p and information exponent k.

Interpretation: f_* focuses on the most prominent directions of the input features.

- Larger spike (SNR) θ $\Rightarrow$ easier problem.

Question: How large should θ be, in order for (i) kernel ridge regression, and (ii) neural network trained by GD, to *learn* f_* *in the proportional regime*?

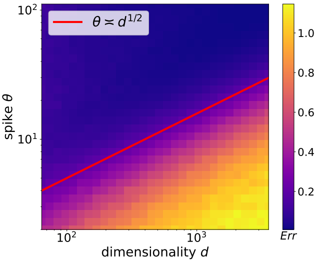
Theorem ([BES+23] Necessary and Sufficient Conditions for KRR)

Given $\ell \in \mathbb{N}$, suppose the spike magnitude satisfies

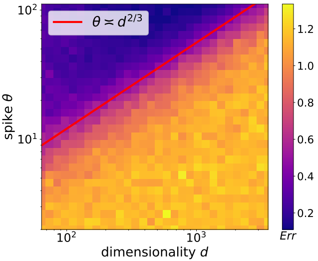
$$\theta \asymp d^{\gamma} \quad \text{for} \quad \gamma \in \left(1 - {}^{1}/_{\ell}, 1 - {}^{1}/_{\ell+1}\right),$$

Then as $n, d \to \infty, n/d \to \psi$, with probability 1, the prediction risk of KRR satisfies

$$\mathcal{R}(\hat{f}_{\mathrm{ker}}) - \|P_{>\ell} f_*\|_{L^2}^2 = o(1).$$



p = 2.



p = 3.

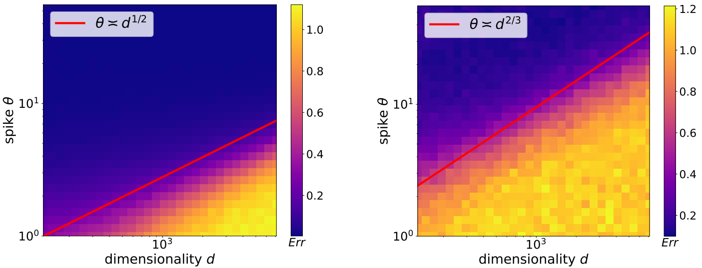
Representation Learning via One Gradient Step

Theorem ([BES+23] Sufficient Condition for NN+GD)

GD-trained two-layer ReLU network with width $N = \Omega(d^{\varepsilon})$ can learn f_* with ***degree p*** *and* ***information exponent k*** *in the proportional regime if*

$$\theta = \omega\left(d^{1-\frac{1}{k}}\right).$$

Observation: required SNR θ does not depend on the *highest degree p*.



k = 2. k = 3.

Neural Network Learnability (sketch)

- **Hermite expansion**. Recall $\boldsymbol{\Sigma} = \boldsymbol{I} + \theta \boldsymbol{\beta}_* \boldsymbol{\beta}_*^\top$, the population gradient is given as

$$
\begin{aligned}
&\mathbb{E}[\boldsymbol{x}\sigma'(\langle \boldsymbol{x}, \boldsymbol{w}\rangle + b) f_*(\boldsymbol{x})] \\
=&(1+\theta)^{-1/2}\boldsymbol{\Sigma}\boldsymbol{\beta}_* \cdot \mathbb{E}\left[\sigma_*'\left((1+\theta)^{-1/2}\langle \boldsymbol{x}, \boldsymbol{\beta}_*\rangle\right)\sigma'(\langle \boldsymbol{x}, \boldsymbol{w}\rangle + b)\right] + \boldsymbol{w}\cdot \mathbb{E}[\ldots] \\
=&\sqrt{1+\theta}\boldsymbol{\beta}_* \cdot \sum_{i=0}^{\infty} (i+1)^2 \alpha_{i+1}^{b} \alpha_{i+1}^{*} \langle \boldsymbol{w}, \sqrt{1+\theta}\boldsymbol{\beta}_*\rangle^i + \ldots
\end{aligned}
$$

  - **Observation 1:** the spike in $\boldsymbol{\Sigma}$ amplifies the gradient in the direction of $\boldsymbol{\beta}_*$.
  - **Observation 2:** bias units "diversify" the nonlinearity σ.

- **Gradient concentration**. To achieve nontrivial concentration when $n \asymp d$,

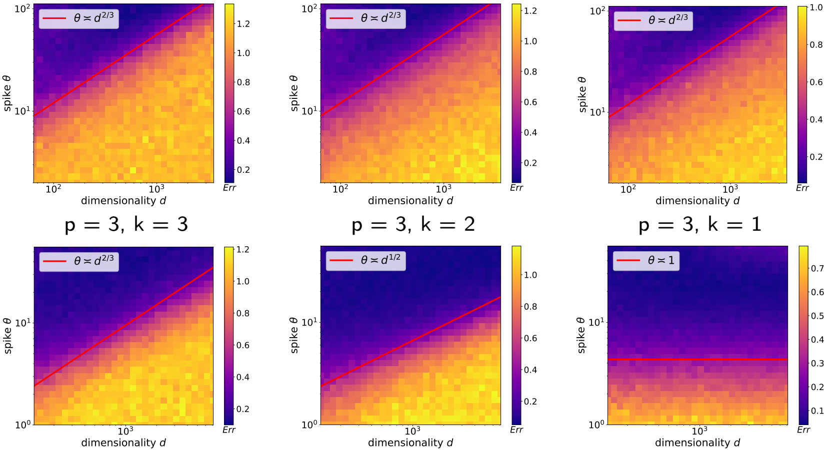
$$\theta = \Omega\left(d^{1-\frac{1}{k}}\right).$$

- **Univariate approximation**. Random bias units to approximate the link σ_*:

$$f_{\mathrm{NN}}(\boldsymbol{x}) = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} a_i \sigma(\boldsymbol{x}^\top \boldsymbol{w}_i + b_i), \quad b_i \sim \mathcal{N}(0,1).$$

Comparing KRR and NN

$k \leq p$ by definition $\implies$ *neural network + gradient descent* (**bottom**) can adapt to low-dimensional structure more efficiently than *kernel method* (**top**).

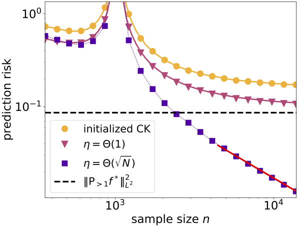


Summary: Learning in the Proportional Regime

Isotropic $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I})$

Spike emerges in updated weights of NN, which improves the performance.

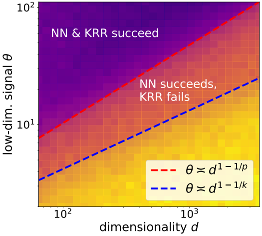
- ❐ KRR only learns *linear* f_* $(p = 1)$
- ❐ NN can learn *nonlinear* f_*, but requires nondegeneracy $(k = 1)$



Anisotropic $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I} + \theta \boldsymbol{\beta}_* \boldsymbol{\beta}_*^\top)$

Spike in the input data improves the performance of both kernel and NN.

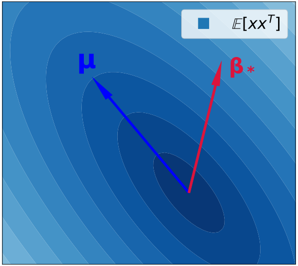
- ❐ KRR: $\theta = \Omega\left(d^{1-\frac{1}{p}}\right)$ *necessary*.
- ❐ NN: $\theta = \omega\left(d^{1-\frac{1}{k}}\right)$ *sufficient*.



Setting: Beyond Perfect Alignment?

Question: what happens if we don't have perfect alignment, i.e. $\boldsymbol{\beta}_* \neq \boldsymbol{\mu}$?

- Problem cannot be solved by PCA + fitting f_* on the top principal component.



❒ **Spiked data:** $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I} + \theta \boldsymbol{\mu}\boldsymbol{\mu}^\top)$.

❒ **Misaligned teacher:**
$f_*(\boldsymbol{x}) = \sigma_*\left(\frac{1}{\sqrt{1+\theta\langle\boldsymbol{\mu},\boldsymbol{\beta}_*\rangle^2}}\langle \boldsymbol{x}, \boldsymbol{\beta}_*\rangle\right)$, with degree p and information exponent k.

Interpretation: f_* is *partially captured* by the most prominent directions of input features.

Spike-target alignment: $\langle \boldsymbol{\mu}, \boldsymbol{\beta}_*\rangle \asymp d^{-\gamma_1}$, **Spike magnitude**: $\theta \asymp d^{\gamma_2}$.

Remark: We take $\gamma_1 \in [0, 1/2]$, and $\gamma_2 \in [0, 1]$.

Insufficiency of One Gradient Step

First gradient step: denote $\kappa = 1 + \theta\langle\boldsymbol{\mu}, \boldsymbol{\beta}_*\rangle^2$, (ignoring the bias terms)

$$\mathbb{E}[\boldsymbol{x}\sigma'(\langle\boldsymbol{x}, \boldsymbol{w}\rangle)f^*(\boldsymbol{x})] = \kappa^{-1/2}\boldsymbol{\Sigma}\boldsymbol{\beta}_* \cdot \mathbb{E}\left[\sigma_*'\left(\kappa^{-1/2}\langle\boldsymbol{x}, \boldsymbol{\beta}_*\rangle\right)\sigma'(\langle\boldsymbol{x}, \boldsymbol{w}\rangle)\right] + \ldots$$
$$= (\boldsymbol{\beta}_* + \langle\boldsymbol{\beta}_*, \boldsymbol{\mu}\rangle\theta \cdot \boldsymbol{\mu}) \cdot \kappa^{-1/2}\mathbb{E}_{\boldsymbol{x}}\left[f_*'(x)\sigma'(x, \boldsymbol{w})\right] + \ldots$$

- ☹ One gradient step does not find the direction of f_*.
- ☺ When $\langle\boldsymbol{\mu}, \boldsymbol{\beta}_*\rangle \asymp d^{-\gamma_1}$ is nontrivial, i.e., $\gamma_1 < 1/2$, the first GD step provides "**warm-start**" to *subsequent gradient updates*.

Goal: characterize the sample complexity[2] of feature learning under varying *Spike-target alignment*: $\langle\boldsymbol{\mu}, \boldsymbol{\beta}_*\rangle \asymp d^{-\gamma_1}$ *Spike magnitude*: $\theta \asymp d^{\gamma_2}$.

Question: what is the suitable gradient dynamics for this setting?

[2] We no longer restrict ourselves to the proportional asymptotic limit.

Algorithm: Spherical Gradient Flow?

Simplification – one-neuron dynamics. Consider $\boldsymbol{w}_0 = \boldsymbol{w}_1 = \dots \boldsymbol{w}_N$ randomly initialized from unit sphere: $f^t(\boldsymbol{x}) = \sigma(\langle \boldsymbol{x}, \boldsymbol{w}^t \rangle)$.

Candidate I – spherical gradient flow [Ben Arous et al. 2021] [Bietti et al. 2022]:

$$\mathrm{d}\boldsymbol{w}^t = -\nabla^S \mathcal{R}(f^t)\,\mathrm{d}t, \quad \nabla^S \mathcal{R}(f^t) := (\boldsymbol{I} - \boldsymbol{w}^t \boldsymbol{w}^{t\top}) \nabla_{\boldsymbol{w}} \mathcal{R}(f^t).$$

Proposition ([MWS+23] Failure of Spherical Gradient, *informal*)

Consider the perfectly aligned setting $\boldsymbol{\beta}_* = \boldsymbol{\mu}$. *Then for the* ***population*** *dynamics,*

$$\sup_{t \geq 0} \left| \langle \boldsymbol{w}^t, \boldsymbol{\beta}_* \rangle \right| \lesssim d^{-1/2},$$

when $\theta \asymp d^{\gamma_2}, \gamma_2 \in \left(0, d^{1 - \frac{1}{k-1}}\right)$, *with probability* 0.99 *over the random initialization.*

Repulsive force: $\mathbb{E}_x[f^t(x)^2]$ grows with $\left| \langle \boldsymbol{w}^t, \boldsymbol{\beta}_* \rangle \right|$, which may *prevent alignment.*

$$\mathcal{R}(f) = \mathbb{E}_{\boldsymbol{x}}[f_*(\boldsymbol{x})^2] - 2 \underbrace{\mathbb{E}_{\boldsymbol{x}}[f_*(\boldsymbol{x}) f(\boldsymbol{x})]}_{\text{correlation ☺}} + \underbrace{\mathbb{E}_{\boldsymbol{x}}[f(\boldsymbol{x})^2]}_{\text{repulsion ☹}}$$

Algorithm: Normalized Gradient Flow

Candidate II – normalized gradient flow: $f^t(\boldsymbol{x}) = \sigma\left(\frac{\langle \boldsymbol{x}, \boldsymbol{w}^t\rangle}{\|\boldsymbol{\Sigma}^{1/2}\boldsymbol{w}^t\|}\right)$,

$$\mathrm{d}\boldsymbol{w}^t = -\eta(\boldsymbol{w}^t)\nabla_{\boldsymbol{w}}\mathcal{R}(f^t)\,\mathrm{d}t, \quad \eta(\boldsymbol{w}^t) = \langle \boldsymbol{w}, \boldsymbol{\Sigma}\boldsymbol{w}\rangle.$$

Intuition: $\mathbb{E}_{\boldsymbol{x}}[f^t(\boldsymbol{x})^2] = \mathbb{E}_{z\sim\mathcal{N}(0,1)}[\sigma(z)^2] \Rightarrow$ objective reduced to *correlation loss* ☺

- Resembles *batch normalization*!

Algorithm 1: Gradient-based training for two-layer neural network

empirical gradient flow on *first-layer*

$$\mathrm{d}\boldsymbol{w}^t = -\eta(\boldsymbol{w}^t)\hat{\boldsymbol{\Sigma}}^{-1}\nabla_{\boldsymbol{w}}\mathcal{R}_n(f^t)\,\mathrm{d}t, \quad \boldsymbol{w}^0 \sim \mathrm{Unif}(\mathbb{S}^{d-1}).$$

ridge regression for *second-layer*

$$\hat{\boldsymbol{a}} \leftarrow \mathrm{argmin}_{\boldsymbol{a}}\left\{\frac{1}{n}\sum_{j=1}^n \left(y_j - \langle \boldsymbol{\phi}_j, \boldsymbol{a}\rangle\right)^2 + \lambda\|\boldsymbol{a}\|^2\right\}, \quad [\boldsymbol{\phi}_j]_i := \frac{1}{\sqrt{N}}\sigma\left(\left\langle \boldsymbol{x}_j, \boldsymbol{w}_i^t\right\rangle + b_i\right).$$

return prediction function $\hat{f}(\boldsymbol{x}) = \frac{1}{\sqrt{N}}\sum_{i=1}^N \hat{a}_i\sigma\left(\left\langle \boldsymbol{x}, \boldsymbol{w}_i^t\right\rangle + b_i\right)$

Sample & Runtime Complexity

Recall $\langle \boldsymbol{\mu}, \boldsymbol{\beta}_* \rangle \asymp d^{-\gamma_1}$ and $\theta \asymp d^{\gamma_2}$, with $\gamma_1 \in [0, 1/2]$ and $\gamma_2 \in [0, 1]$.

Theorem ([MWS+23] Complexity of Empirical Gradient Flow)

Two-layer ReLU network learns f_* with ***information exponent k*** *and width $m \asymp \varepsilon^{-1}$, if the sample complexity satisfies*[3]

$$n \gtrsim \begin{cases} d\left(d^{k-1} \vee \varepsilon^{-2}\right) & 0 \le \gamma_2 < \gamma_1, \\ d\left(d^{(k-1)(1-2(\gamma_2-\gamma_1))} \vee \varepsilon^{-2}\right) & \gamma_1 < \gamma_2 < 2\gamma_1, \\ d\left(d^{(k-1)(1-\gamma_2)} \vee \varepsilon^{-2}\right) & 2\gamma_1 < \gamma_2 < 1, \end{cases}$$

and the gradient flow runtime satisfies $T \asymp \tau_k(\delta_0) + \ln(1/\varepsilon)$, where

$$\tau_k(z) := \begin{cases} 1 & k = 1 \\ \ln(1/z) & k = 2 \\ (1/z)^{k-2} & k > 2 \end{cases} \quad \text{and} \quad \delta_0 = \begin{cases} d^{-1/2} & 0 \le \gamma_2 < \gamma_1 \\ d^{\gamma_2-\gamma_1-1/2} & \gamma_1 < \gamma_2 < 2\gamma_1 \\ d^{(\gamma_2-1)/2} & 2\gamma_1 < \gamma_2 < 1 \end{cases}.$$

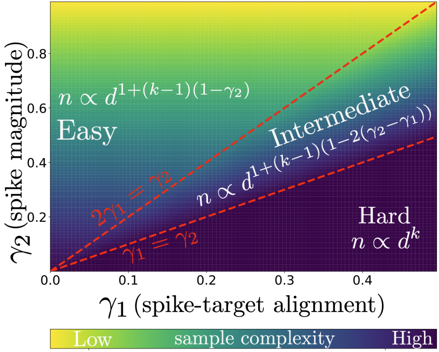
[3] Requires an assumption on the link function: $\zeta(\omega) = \sum_{j \ge k} j\alpha_j^* \alpha_j\, \omega^{j-1} \ge c\, \omega^{k-1}, \forall \omega \in (0,1)$, which may be removed by introducing random bias units.

Interplay between Spike Magnitude and Alignment

Recall $\langle \boldsymbol{\mu}, \boldsymbol{\beta}_* \rangle \asymp d^{-\gamma_1}$ and $\theta \asymp d^{\gamma_2}$, with $\gamma_1 \in [0, 1/2]$ and $\gamma_2 \in [0, 1]$.

Interpretation of Rates:

- $\gamma_1 = 0$: perfect alignment puts us in the "easy" regime.
- $\gamma_1 = 0.5$: two independent $\boldsymbol{\mu}$ and $\boldsymbol{\beta}_*$ on unit sphere.
- $\gamma_1 \in (0, 0.5)$: problem gets easier for larger γ_2.



Theorem ([Donhauser et al. 2021] KRR lower bound, *informal*)

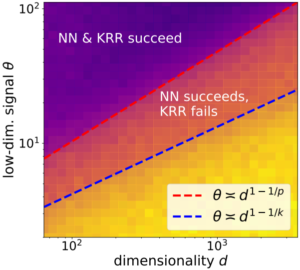
Rotationally invariant kernels require at least $n \asymp d^{\Theta((1-\gamma_2)p)}$ *samples to learn* f_*.

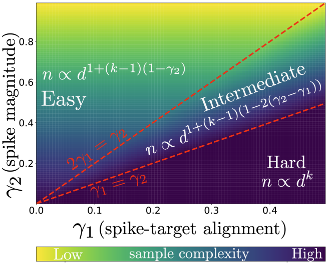
Conclusion: Learning under Structured Data

So far: learning *single-index model* under *spiked covariance* data.

$$\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I} + \theta \boldsymbol{\mu}\boldsymbol{\mu}^\top),\ f_*(\boldsymbol{x}) = \sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}_* \rangle), \quad \text{where } \langle \boldsymbol{\beta}_*, \boldsymbol{\mu} \rangle \asymp d^{-\gamma_1},\ \theta \asymp d^{\gamma_2}.$$

- **Perfectly aligned setting** ($\boldsymbol{\beta}_* = \boldsymbol{\mu}$).
  - Precise analysis for KRR; upper bound for two-layer NN + one GD step.
- **Partially aligned setting** ($\boldsymbol{\beta}_* \neq \boldsymbol{\mu}$).
  - Sample complexity analysis of normalized gradient flow.





Beyond "Narrow" NNs: the Mean-field Regime

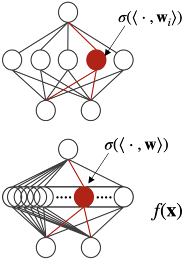
"Blessing" of overparameterization: recall that

$$\mathbb{E}[\nabla_{\boldsymbol{w}_i}\mathcal{L}(f_{\mathsf{NN}})] \approx \boldsymbol{\beta}_* \cdot \sum_{i=0}^{\infty}(i+1)^2\alpha_{i+1}\alpha_{i+1}^*\langle \boldsymbol{w}_i, \boldsymbol{\beta}_*\rangle^i + \ldots$$

☺ If the NN is *sufficiently wide*, there exists some $\boldsymbol{w}_i$ with $\langle \boldsymbol{w}_i, \boldsymbol{\beta}_*\rangle \gg d^{-1/2}$.

☹ Required width may be *exponential* in the dimensionality d.

Mean-field limit : *infinite-width* two-layer neural network



$$f(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\sigma(\langle \mathbf{x}, \mathbf{w}_i\rangle)$$

↓ $N \to \infty$

$$f(\mathbf{x}) = \mathbb{E}_p\left[\sigma(\langle \mathbf{x}, \mathbf{w}\rangle)\right] = \int \sigma(\langle \mathbf{x}, \mathbf{w}\rangle)p(\mathbf{w})\mathrm{d}\mathbf{w}$$

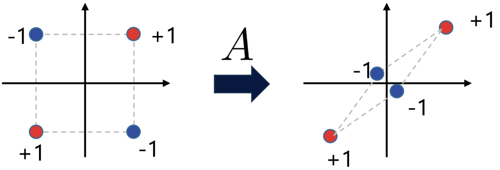
For convex loss L, learning is

- non-convex w.r.t. $\boldsymbol{w}_i$
- convex w.r.t. distribution p

Perspective: study optimization in the space of measures (Wasserstein gradient flow, etc.)

Classifying Sparse Parity Functions

Anisotropic k-parity: $x = Az,\ y = \mathrm{sign}\left(\prod_{i\in I_k} z_i\right),\ z_i \overset{i.i.d.}{\sim} \mathrm{Unif}\left(\{\pm 1/\sqrt{d}\}\right).$



- Analogous to single-index f_* with *information exponent k*.
- When $A = I$ (isotropic), CSQ lower bound $n \asymp d^{k-1}$.
- $k = 2 \Rightarrow$ XOR problem.

Example - spiked covariance: for $i \in I_k, j \notin I_k$ we have $\frac{x_i}{x_j} \asymp d^{\alpha/2}$.

Theorem ([SWO+23] Mean-field Learning of Anisotropic Parity)

Two-layer NN optimized by noisy gradient descent learns k-parity with

$$n = \Theta(d^{1-\alpha}),\quad N = \exp(d^{1-\alpha}),\quad t = \exp(d^{1-\alpha}).$$

Observation: sample complexity *independent* of information exponent (leap) k.

Analysis: Mean-field Langevin Dynamics

Mean-field Langevin dynamics. Given convex $F : \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}$,

$$\mathrm{d}X_t = -\nabla \frac{\delta F(\mu_t)}{\delta \mu}(X_t)\,\mathrm{d}t + \sqrt{2\lambda}\,\mathrm{d}W_t, \quad \mu_t = \mathrm{Law}(X_t).$$

- Wasserstein gradient flow that minimizes $\min_{\mu \in \mathcal{P}_2} \{F(\mu) + \lambda \mathrm{Ent}(\mu)\}$.

☺ **Exponential convergence** in the infinite-width & continuous-time limit.

- [NWS22] *Convex analysis of the mean-field Langevin dynamics*
- Chizat 22. *Mean-field langevin dynamics: exponential convergence and annealing*

☺ **Uniform-in-time propagation of chaos** at any *fixed* temperature λ.

- Chen et al. 23. *Uniform propagation of chaos for mean-field Langevin dynamics*
- [SWN23] *Mean-field Langevin dynamics: time and space discretization, stochastic gradient, and variance reduction*

☹ **Logarithmic Sobolev constant** depends *exponentially* on λ.

- Anneal $\lambda \asymp d^{-1}$ to learn low-dimensional f_* $\Rightarrow$ exponential computation...

Question: Poly-time learning guarantees for an interesting class of f_*?

References

Thank you! Happy to take questions :)

- Ghorbani et al., 2020. *When do neural networks outperform kernel methods?*
- Hu and Lu, 2020. *Universality laws for high-dimensional learning with random features*.
- Ben Arous et al., 2021. *Stochastic gradient descent on non-convex losses from high-dimensional inference*.
- Bartlett et al., 2021. *Deep learning: a statistical viewpoint*.
- Refinetti et al., 2021. *Classifying high-dimensional Gaussian mixtures: where kernel methods fail and neural networks succeed*.
- Abbe et al., 2022. *The merged-staircase property: a necessary and nearly sufficient condition for SGD learning of sparse functions on two-layer neural networks*.
- Damien et al., 2022. *Neural networks can learn representations with gradient descent*.
- Bietti et al., 2022. *Learning single-index models with shallow neural networks*.
- Berthier et al., 2023. *Learning time-scales in two-layers neural networks*.
- Abbe et al. 2023. *SGD learning on neural networks: leap complexity and saddle-to-saddle dynamics*.