Learning features with two-layer neural networks, one step at a time

Bruno Loureiro
@ CSD, DI-ENS & CNRS

brloureiro@gmail.com

CIMI Workshop “Recent developments beyond classical regimes in statistical learning”, 04.11.2024

How Two-Layer Neural Networks Learn, One (Giant) Step at a Time

Yatin Dandi[1,3], Florent Krzakala[1], Bruno Loureiro[2], Luca Pesce[1], and Ludovic Stephan[1]

arXiv: 2305.18270
(JMLR)

Asymptotics of feature learning in two-layer networks after one gradient-step

Hugo Cui[1], Luca Pesce[2], Yatin Dandi[2,1], Florent Krzakala[2], Yue M. Lu[3],
Lenka Zdeborová[1], and Bruno Loureiro[4]

arXiv: 2402.04980
(ICML 2024)

A Random Matrix Theory Perspective on the Spectrum of Learned Features and Asymptotic Generalization Capabilities

Yatin Dandi[1,2], Luca Pesce[2], Hugo Cui[1,5], Florent Krzakala[2], Yue M. Lu[3], and Bruno Loureiro[4]

arXiv: 2410.18938

How Two-Layer Neural Networks Learn, One (Giant) Step at a Time

Yatin Dandi[1,3], Florent Krzakala[1], Bruno Loureiro[2], Luca Pesce[1], and Ludovic Stephan[1]

arXiv: 2305.18270
(JMLR)

Asymptotics of feature learning in two-layer networks after one gradient-step

Hugo Cui[1], Luca Pesce[2], Yatin Dandi[2,1], Florent Krzakala[2], Yue M. Lu[3],
Lenka Zdeborová[1], and Bruno Loureiro[4]

arXiv: 2402.04980
(ICML 2024)

A Random Matrix Theory Perspective on the Spectrum of Learned Features and Asymptotic Generalization Capabilities

Yatin Dandi[1,2], Luca Pesce[2], Hugo Cui[1,5], Florent Krzakala[2], Yue M. Lu[3], and Bruno Loureiro[4]

arXiv: 2410.18938

Why deep neural networks work better than other machine learning methods?

Deep neural networks (DNNs) often outperform other machine learning methods in many applications due to several key factors:

1. Representation Learning:

- **Automatic Feature Extraction:** One of the most significant advantages of deep neural networks is their ability to automatically learn and extract features from raw data. In traditional machine learning methods, feature engineering (manually designing features) is a critical step, and the quality of the features significantly affects performance. DNNs can learn hierarchical feature representations directly from the data, capturing complex patterns that are difficult to manually design.
- **Hierarchical Representations:** DNNs learn features at multiple levels of abstraction. The lower layers capture low-level features (e.g., edges in images), and higher layers capture more abstract concepts (e.g., object parts or entire objects). This hierarchical feature learning allows DNNs to handle the complexity of high-dimensional data effectively.

How do we define this mathematically?

How do we define this mathematically?

Can we quantify impact in generalisation?

How do we define this mathematically?

Can we quantify impact in generalisation?

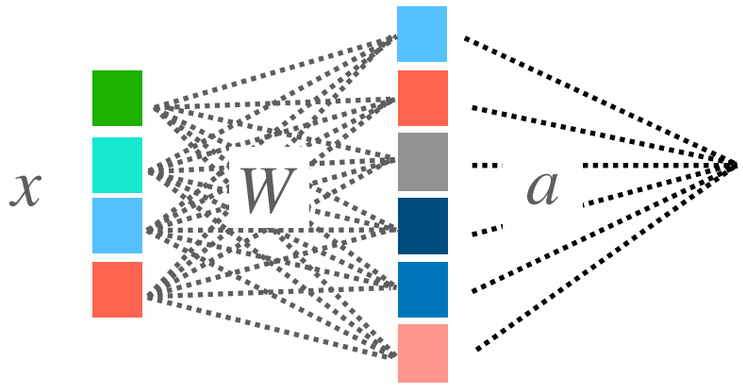
<u>This talk:</u> Exact picture in a simple setting:

One GD step in a 2-layer neural net
with random data

Setting

Our protagonist - 2 layer NNs:

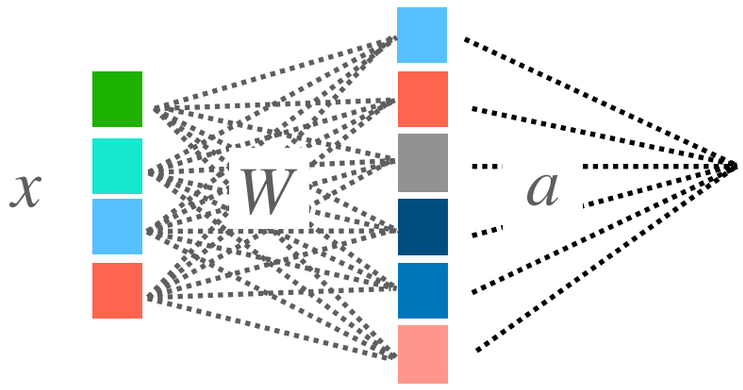
$$f(x; a, W) = \frac{1}{\sqrt{p}} \sum_{k=1}^{p} a_k \sigma\left(\langle w_k, x \rangle\right)$$



Setting

Our protagonist - 2 layer NNs:

$$f(x; a, W) = \frac{1}{\sqrt{p}} \sum_{k=1}^{p} a_k \sigma\left(\langle w_k, x \rangle\right)$$



We assume training data $\mathscr{D} = \{(x_i, y_i) \in \mathbb{R}^{d+1} : i \in [n]\}$ is drawn from:

$$y_i = f_\star(x_i) + z_i$$

$$x_i \sim \mathcal{N}(0, I_d/d) \qquad z_i \sim \mathcal{N}(0, \Delta)$$

What can we learn initialisation $W = W^0$?

(by training only 2nd layer weights $a \in \mathbb{R}^p$)

What can we learn initialisation $W = W^0$?

(by training only 2nd layer weights $a \in \mathbb{R}^p$)

$$\hat{a}_\lambda(X, y) = \underset{a \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \langle a, \sigma(W^0 x_i) \rangle^2 + \lambda ||a||_2^2$$

Random features model

1. When $p \to \infty$, W stays close to W^0 (lazy regime)

[Jacot, Gabriel, Hongler '18;
Chizat, Bach '19;
Neal '94; Lee et al. '19]

Random features model

1. When $p \to \infty$, W stays close to W^0 (lazy regime)

[Jacot, Gabriel, Hongler '18;
Chizat, Bach '19;
Neal '94; Lee et al. '19]

2. This can be seen as the approximation for a kernel method (universal approximators)

$$K_{\mathrm{RF}}(x, x') = \mathbb{E}_{w_0}\left[\sigma\left(\langle w^0, x\rangle\right)\sigma\left(\langle w^0, x'\rangle\right)\right] \approx \frac{1}{p}\sum_{k=1}^{p}\sigma\left(\langle w_k^0, x\rangle\right)\sigma\left(\langle w_k^0, x'\rangle\right)$$

[Retch, Raimi 2007]

Random features model

1. When $p \to \infty$, W stays close to W^0 (lazy regime)

[Jacot, Gabriel, Hongler '18;
Chizat, Bach '19;
Neal '94; Lee et al. '19]

2. This can be seen as the approximation for a kernel method (universal approximators)

$$K_{\mathrm{RF}}(x, x') = \mathbb{E}_{w_0}\left[\sigma\left(\langle w^0, x\rangle\right)\sigma\left(\langle w^0, x'\rangle\right)\right] \approx \frac{1}{p}\sum_{k=1}^{p}\sigma\left(\langle w_k^0, x\rangle\right)\sigma\left(\langle w_k^0, x'\rangle\right)$$

[Retch, Raimi 2007]

🤔 What can we learn with that?

Louart et al., '18; Mei, Montanari '19; Ghorbani, Mei, Misiakiewicz, Montanari '19, '20, '21; Gerace, **BL**, Krzakala, Mézard, Zdeborová '20; Goldt, **BL**, Reeves, Krzakala, Mézard, Zdeborová '21 Dhiffalah & Lu '20; Hu & Lu '20; Liang, Sur '20; Jacot, Simsek, Spadaro, Hongler, Gabriel '20; **BL**, Gerbelot, Refinetti, Sicuro, Krzakala '22; Mei, Misiakiewicz, Montanari '22; Fan, Wang 2020; Liao et al., '21; Schröder, Cui, Dmitriev, **BL** '23, 24; Defilippis, **BL**, Misiakiewicz 24.

Limitations of RF

Theorem [Mei, Misiakiewicz, Montanari ’22, informal]:

For isotropic data (e.g. $x \sim \mathrm{Unif}(\mathbb{S}^{d-1})$, with $n, p = \Theta(d^\kappa)$ one can learn at best a polynomial approximation of degree κ of the target $f_\star(x)$

$$\mathbb{E}\,||f_\star(x) - f(x; \hat{a}_\lambda, W^0)||_2^2 = ||P_{>\kappa} f_\star||_{L_2}^2 + o_d(1)$$

Limitations of RF

Theorem [Mei, Misiakiewicz, Montanari '22, informal]:

For isotropic data (e.g. $x \sim \mathrm{Unif}(\mathbb{S}^{d-1})$), with $n, p = \Theta(d^\kappa)$ one can learn at best a polynomial approximation of degree κ of the target $f_\star(x)$

$$\mathbb{E}\,||f_\star(x) - f(x; \hat{a}_\lambda, W^0)||_2^2 = ||P_{\leq \kappa} f_\star||_{L_2}^2 + o_d(1)$$

In particular, for $n, p = \Theta(d)$, can learn at best a linear approximation of $f_\star$

$$f_\star(x) = \langle \theta_\star, x \rangle + f_{NL}(x)$$

Limitations of RF

Theorem [Mei, Misiakiewicz, Montanari '22, informal]:

For isotropic data (e.g. $x \sim \mathrm{Unif}(\mathbb{S}^{d-1})$), with $n, p = \Theta(d^\kappa)$ one can learn at best a polynomial approximation of degree κ of the target $f_\star(x)$

$$\mathbb{E}\,||f_\star(x) - f(x; \hat{a}_\lambda, W^0)||_2^2 = ||P_{\leq \kappa} f_\star||_{L_2}^2 + o_d(1)$$

In particular, for $n, p = \Theta(d)$, can learn at best a linear approximation of $f_\star$

$$f_\star(x) = \langle \theta_\star, x \rangle + f_{NL}(x)$$

Intuition:

$$\sigma(\langle w^0, x \rangle) = \mu_0 + \mu_1 \langle w^0, x \rangle + \sum_{\alpha \geq 2} \frac{\mu_\alpha}{\alpha!} \mathrm{He}_\alpha(\langle w^0, x \rangle)$$

$$\mu_\alpha = \mathbb{E}[\mathrm{He}_\alpha(z)\sigma(z)]$$

Limitations of RF

Theorem [Mei, Misiakiewicz, Montanari '22, informal]:

For isotropic data (e.g. $x \sim \mathrm{Unif}(\mathbb{S}^{d-1})$), with $n, p = \Theta(d^\kappa)$ one can learn at best a polynomial approximation of degree κ of the target $f_\star(x)$

$$\mathbb{E}\,||f_\star(x) - f(x; \hat{a}_\lambda, W^0)||_2^2 = ||P_{\leq \kappa} f_\star||_{L_2}^2 + o_d(1)$$

In particular, for $n, p = \Theta(d)$, can learn at best a linear approximation of $f_\star$

$$f_\star(x) = \langle \theta_\star, x \rangle + f_{NL}(x)$$

Intuition:

$$\sigma(\langle w^0, x \rangle) = \mu_0 + \mu_1 \langle w^0, x \rangle + \sum_{\alpha \geq 2} \frac{\mu_\alpha}{\alpha!} \mathrm{He}_\alpha(\langle w^0, x \rangle)$$

$$= \Theta(d^{-\alpha/2})$$

$$\mu_\alpha = \mathbb{E}[\mathrm{He}_\alpha(z)\sigma(z)]$$

Limitations of RF

Theorem [Mei, Misiakiewicz, Montanari '22, informal]:

For isotropic data (e.g. $x \sim \mathrm{Unif}(\mathbb{S}^{d-1})$), with $n, p = \Theta(d^\kappa)$ one can learn at best a polynomial approximation of degree κ of the target $f_\star(x)$

$$\mathbb{E}\,||f_\star(x) - f(x; \hat{a}_\lambda, W^0)||_2^2 = ||P_{\leq\kappa} f_\star||_{L_2}^2 + o_d(1)$$

In particular, for $n, p = \Theta(d)$, can learn at best a linear approximation of $f_\star$

$$f_\star(x) = \langle \theta_\star, x \rangle + f_{NL}(x)$$

Intuition:

$$\sigma(\langle w^0, x\rangle) = \mu_0 + \mu_1 \langle w^0, x\rangle + \sum_{\alpha \geq 2} \frac{\mu_\alpha}{\alpha!} \mathrm{He}_\alpha(\langle w^0, x\rangle)$$

$$= \Theta(d^{-\alpha/2})$$

$$\approx \mu_0 + \mu_1 \langle w, x\rangle + \mu_\star \xi$$

$$\mu_\alpha = \mathbb{E}[\mathrm{He}_\alpha(z)\sigma(z)] \qquad \mu_\star = \sqrt{\mathbb{E}[\sigma(z)^2] - \mu_0^2 - \mu_1^2}$$

Gaussian equivalence

Consider the following two ERM problems:

$$\hat{a}_\lambda(X,y) = \underset{a}{\operatorname{argmin}} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \langle a, \sigma(W^0 x_i))^2 + \lambda \,||\, a \,||_2^2$$

$$\hat{a}^G_\lambda(X,y) = \underset{a}{\operatorname{argmin}} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \langle a, \mu_0 1 + \mu_1 W^0 x_i + \mu_\star z_i \rangle)^2 + \lambda \,||\, a \,||_2^2$$

Then, in the limit $d \to \infty$ with $n, p = \Theta(d)$:

Gaussian equivalence principle (GEP)
[Goldt et al. ’19, 20; Mei & Montanari ’19; Hu& Lu ’20]

$$|\, R(\hat{a}_\lambda) - R(\hat{a}^G_\lambda) \,| \to 0$$

Definitions:

Consider the unique fixed point of the following system of equations

$$\begin{cases} \hat{V}_s = \frac{\alpha}{\gamma}\kappa_1^2 \mathbb{E}_{\xi,y}\left[\mathcal{Z}\left(y,\omega_0\right)\frac{\partial_\omega \eta(y,\omega_1)}{V}\right], \\ \hat{q}_s = \frac{\alpha}{\gamma}\kappa_1^2 \mathbb{E}_{\xi,y}\left[\mathcal{Z}\left(y,\omega_0\right)\frac{\left(\eta(y,\omega_1)-\omega_1\right)^2}{V^2}\right], \\ \hat{m}_s = \frac{\alpha}{\gamma}\kappa_1 \mathbb{E}_{\xi,y}\left[\partial_\omega \mathcal{Z}\left(y,\omega_0\right)\frac{(\eta(y,\omega_1)-\omega_1)}{V}\right], \\ \hat{V}_w = \alpha\kappa_\star^2 \mathbb{E}_{\xi,y}\left[\mathcal{Z}\left(y,\omega_0\right)\frac{\partial_\omega \eta(y,\omega_1)}{V}\right], \\ \hat{q}_w = \alpha\kappa_\star^2 \mathbb{E}_{\xi,y}\left[\mathcal{Z}\left(y,\omega_0\right)\frac{\left(\eta(y,\omega_1)-\omega_1\right)^2}{V^2}\right], \end{cases}$$

$$\begin{cases} V_s = \frac{1}{\hat{V}_s}\left(1 - z\ g_\mu(-z)\right), \\ q_s = \frac{\hat{m}_s^2+\hat{q}_s}{\hat{V}_s}\left[1 - 2z g_\mu(-z) + z^2 g'_\mu(-z)\right] \\ \quad -\frac{\hat{q}_w}{(\lambda+\hat{V}_w)\hat{V}_s}\left[-z g_\mu(-z) + z^2 g'_\mu(-z)\right], \\ m_s = \frac{\hat{m}_s}{\hat{V}_s}\left(1 - z\ g_\mu(-z)\right), \\ V_w = \frac{\gamma}{\lambda+\hat{V}_w}\left[\frac{1}{\gamma} - 1 + z g_\mu(-z)\right], \\ q_w = \gamma\frac{\hat{q}_w}{(\lambda+\hat{V}_w)^2}\left[\frac{1}{\gamma} - 1 + z^2 g'_\mu(-z)\right], \\ \quad +\frac{\hat{m}_s^2+\hat{q}_s}{(\lambda+\hat{V}_w)\hat{V}_s}\left[-z g_\mu(-z) + z^2 g'_\mu(-z)\right], \end{cases}$$

$$\begin{cases} \eta(y,\omega) = \underset{x\in\mathbb{R}}{\operatorname{argmin}}\left[\frac{(x-\omega)^2}{2V} + \ell(y,x)\right] \\ \mathcal{Z}(y,\omega) = \int \frac{dx}{\sqrt{2\pi V^0}} e^{-\frac{1}{2V^0}(x-\omega)^2}\delta\left(y - f^0(x)\right) \end{cases}$$

where $V = \kappa_1^2 V_s + \kappa_\star^2 V_w, V^0 = \rho - \frac{M^2}{Q}, Q = \kappa_1^2 q_s + \kappa_\star^2 q_w, M = \kappa_1 m_s, \omega_0 = M/\sqrt{Q}\xi, \omega_1 = \sqrt{Q}\xi$

and $g_\mu$is the Stieltjes transform of $W_0 W_0^T \mu_0 = \mathbb{E}\left[\sigma(z)\right], \mu_1 \equiv \mathbb{E}\left[z\sigma(z)\right], \mu_\star \equiv \mathbb{E}\left[\sigma(z)^2\right] - \mu_0^2 - \mu_1^2$, and $z \sim \mathcal{N}(0,1)$

In the high-dimensional limit:

$$R(\hat{a}_\lambda) = \mathbb{E}_{\lambda,\nu}\left[(f^0(\nu) - \hat{f}(\lambda))^2\right]$$

with $(\nu,\lambda) \sim \mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}\rho & M^\star \\ M^\star & Q^\star\end{pmatrix}\right)$

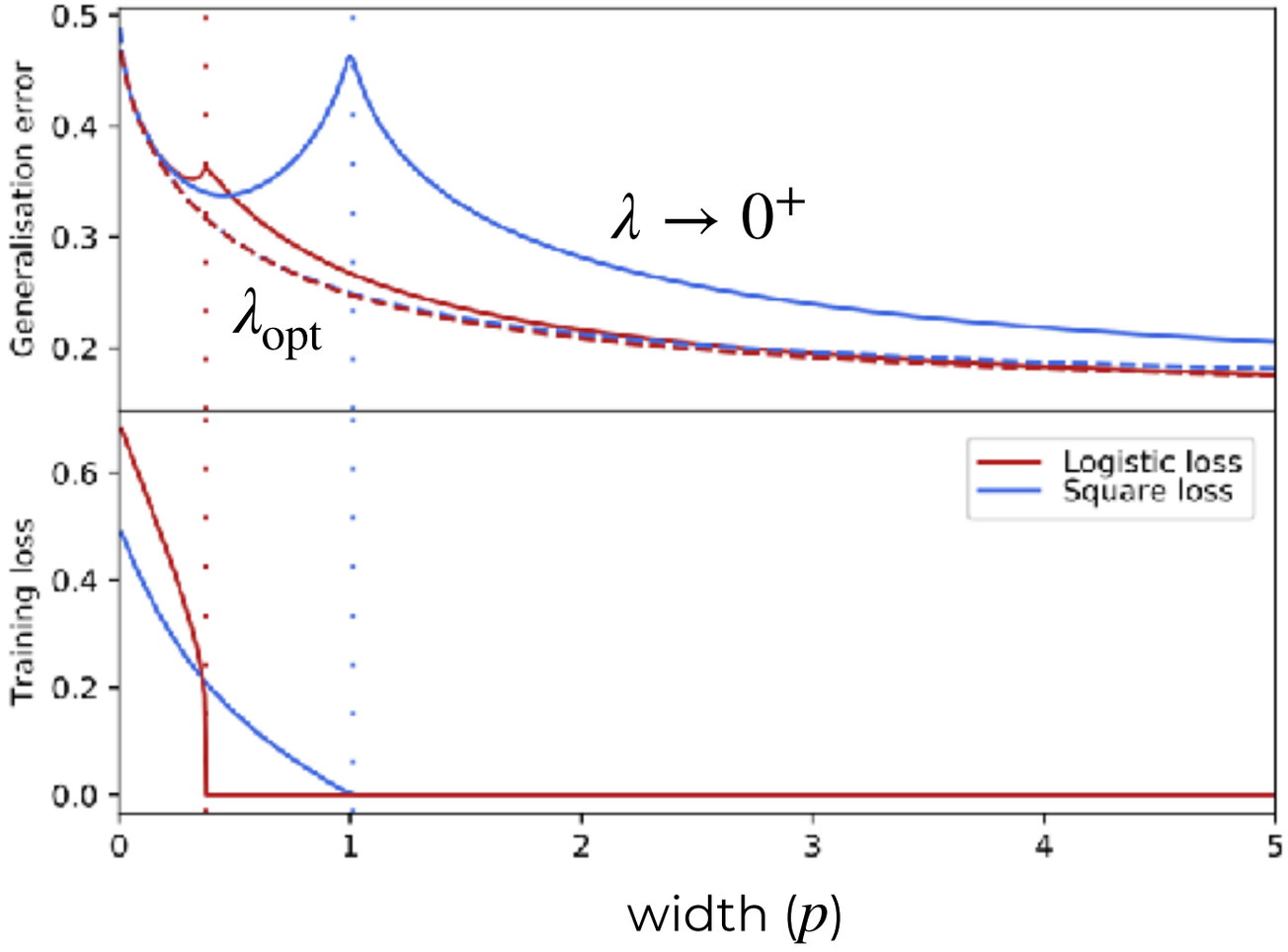
$$\hat{R}_n(\hat{a}_\lambda) = \frac{\lambda}{2\alpha}q_w^\star + \mathbb{E}_{\xi,y}\left[\mathcal{Z}\left(y,\omega_0^\star\right)\ell\left(y,\eta(y,\omega_1^\star)\right)\right]$$

with $\omega_0^\star = M^\star/\sqrt{Q^\star}\xi, \omega_1^\star = \sqrt{Q^\star}\xi$

Gaussian equivalence

Gaussian equivalence principle (GEP)
[Goldt et al. '19; Mei & Montanari '19; Hu& Lu '20]

$$|\, R(\hat{a}_\lambda) - R(\hat{a}^G_\lambda) \,| \to 0$$

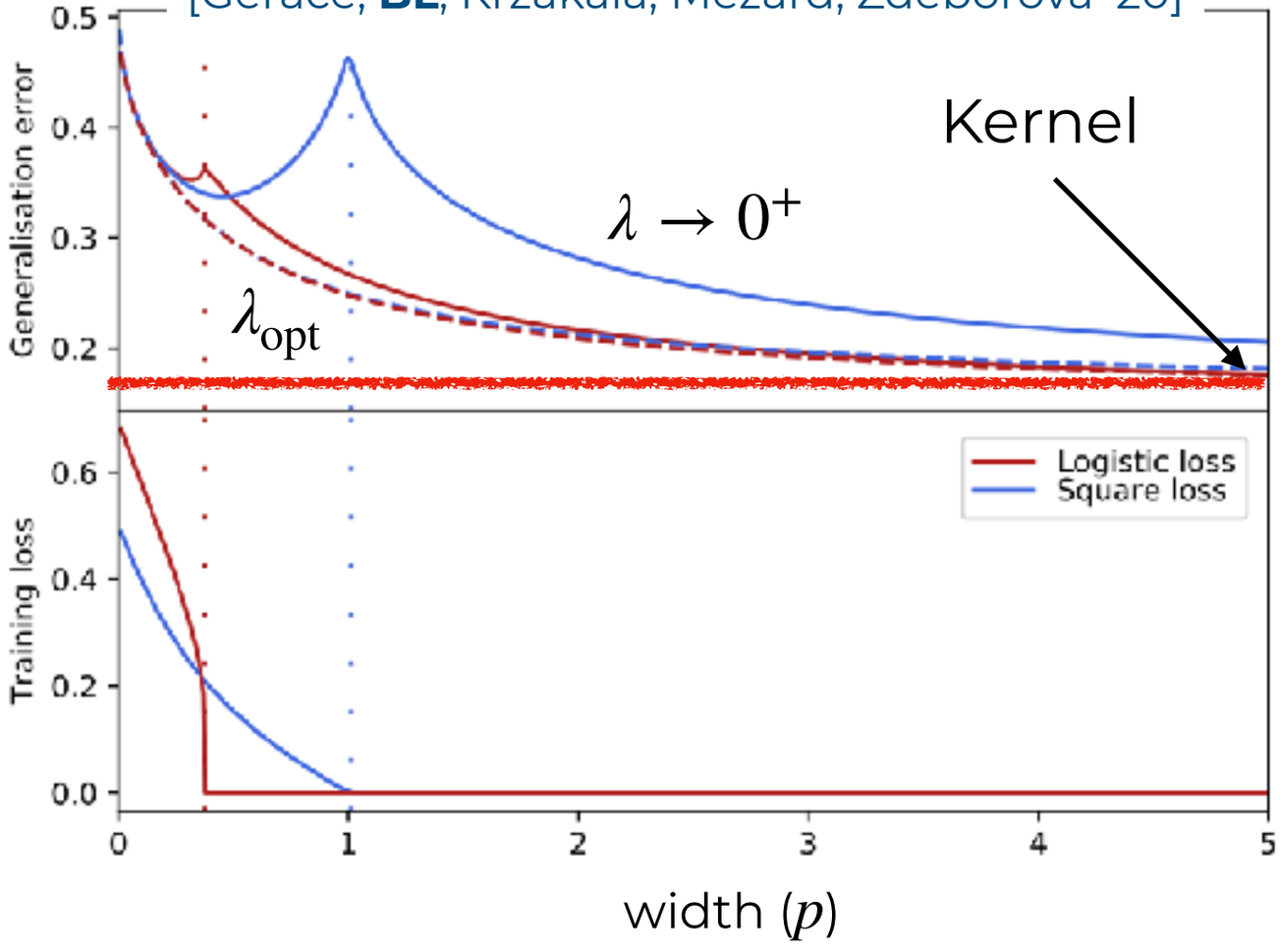


Gaussian equivalence

Gaussian equivalence principle (GEP)
[Goldt et al. '19; Mei & Montanari '19; Hu& Lu '20]

$$|\, R(\hat{a}_\lambda) - R(\hat{a}^G_\lambda)\,| \to 0$$

[Gerace, **BL**, Krzakala, Mézard, Zdeborová '20]



Gaussian universality

↕

Equivalence to a linear model

↕

Limited expressivity

Beyond proportional

What is the minimal number of features p needed to achieve same performance of kernel ($p \to \infty$)?

[Retch, Raimi 2007]: crude $p \geq O(n)$ bound

[Rudi, Rosasco 2017]: improved $p \geq O(\sqrt{n})$ bound

Beyond proportional

What is the minimal number of features p needed to achieve same performance of kernel ($p \to \infty$)?

Dimension-free deterministic equivalents
for random feature regression

Leonardo Defilippis[1], Bruno Loureiro[1], and Theodor Misiakiewicz[2]

May 27, 2024

Abstract

In this work we investigate the generalization performance of random feature ridge regression (RFRR). Our main contribution is a general deterministic equivalent for the test error of RFRR. Specifically, under a certain concentration property, we show that the test error is well approximated by a closed-form expression that only depends on the feature map eigenvalues. Notably, our approximation guarantee is non-asymptotic, multiplicative, and independent of the feature map dimension—allowing for infinite-dimensional features. We expect this deterministic equivalent to hold broadly beyond our theoretical analysis, and we empirically validate its predictions on various real and synthetic datasets. As an application, we derive sharp excess error rates under standard power-law assumptions of the spectrum and target decay. In particular, we provide a tight result for the smallest number of features achieving optimal minimax error rate.

Beyond proportional

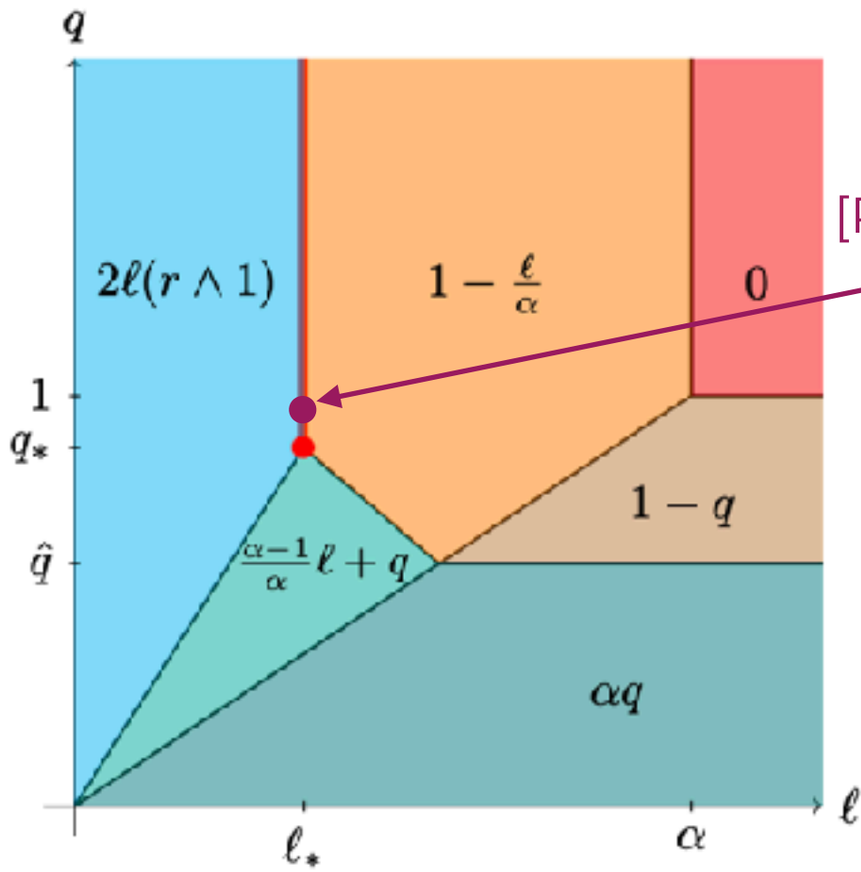
Under source and capacity conditions

$$K(x,x') = \sum_{k\geq 1} \eta_k \varphi_k(x)\varphi_k(x') \qquad f_\star(x) = \sum_{k\geq 1} f_{\star,k}\varphi_k(x)$$

$$p \sim n^q \qquad \eta_k \sim k^{-\alpha}$$

$$\lambda \sim n^{-\ell} \qquad f_{\star,k} = k^{-\alpha r - 1/2}$$

$$\mathbb{E}\,||f(x) - f_\star(x)||^2 \sim n^{-\gamma}$$



[Rudi, Rosasco 2017]

See also [Cui et al. 2021]

Neural scaling law literature [Maloney et al. 2022]

Partial Summary

Kernels/RF are able to learn “anything”,
but they need “a lot” of data.

Partial Summary

Kernels/RF are able to learn “anything”, but they need “a lot” of data.

In particular, with $n, p = \Theta(d)$, only learn linear functions.

Partial Summary

Kernels/RF are able to learn “anything”, but they need “a lot” of data.

In particular, with $n, p = \Theta(d)$, only learn linear functions.

To do better, need to learn features.

What can we learn after one GD step ?

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (y_i - f(x_i; a^0, W^0))^2$$

What can we learn after one GD step ?

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (y_i - f(x_i; a^0, W^0))^2$$

Two flavours of results:

1. Weak learnability: How much W^1 correlates with $f_\star$?

2. Generalisation: How much this improves the error?

What can we learn after one GD step ?

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (y_i - f(x_i; a^0, W^0))^2$$

Two flavours of results:

1. Weak learnability: How much W^1 correlates with $f_\star$?

2. Generalisation: How much this improves the error?

What you learn in one-step of SGD?

Key idea: Hermite tensor decomposition

$$f_\star(x) = \sum_{\alpha \in \mathbb{N}^d} \mu_\alpha h_\alpha(x)$$

What you learn in one-step of SGD?

Key idea: Hermite tensor decomposition

$$f_\star(x) = \sum_{\alpha \in \mathbb{N}^d} \mu_\alpha h_\alpha(x)$$

Allow us to compute the signal component of the gradient:

$$\lim_{d \to \infty} \mathbb{E}[f_\star(x) w_k^1] = ???$$

What you learn in one-step of SGD?

Key idea: Hermite tensor decomposition

$$f_\star(x) = \sum_{\alpha \in \mathbb{N}^d} \mu_\alpha h_\alpha(x)$$

Allow us to compute the signal component of the gradient:

$$\lim_{d \to \infty} \mathbb{E}[f_\star(x) w_k^1] = ???$$

Hardness ≈ targets with no low-frequencies components

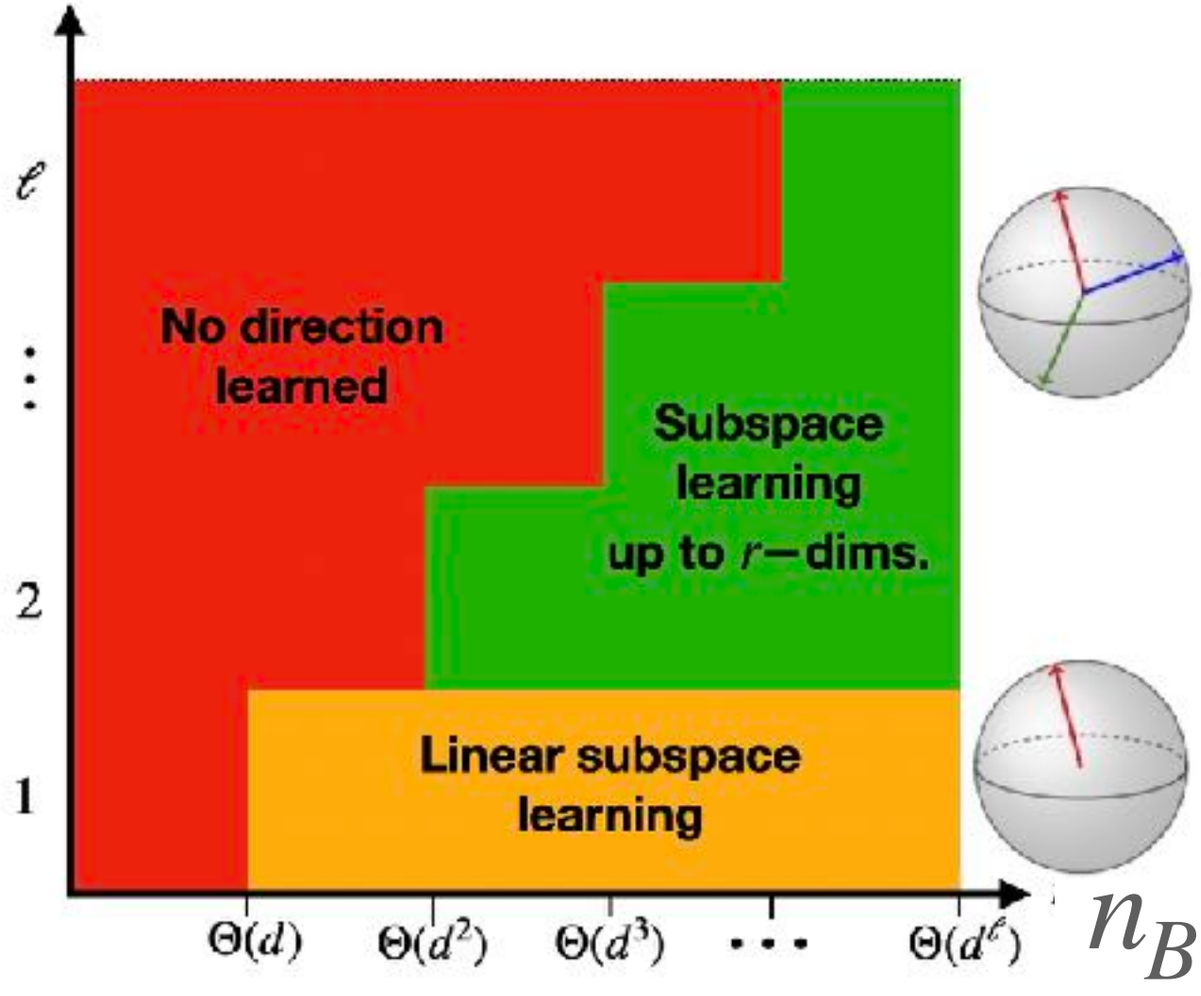
"Leap exponent" ℓ

What you learn in one-step of SGD?

Key idea: Hermite tensor decomposition

$$f_\star(x) = \sum_{\alpha \in \mathbb{N}^d} \mu_\alpha h_\alpha(x)$$

Allow us to compute the signal component of the gradient:



Hardness ≈ targets with no low-frequencies components

"Leap exponent" ℓ

What can we learn after one GD step ?

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (y_i - f(x_i; a^0, W^0))^2$$

Two flavours of results:

1. Weak learnability: How much W^1 correlates with $f_\star$?

2. Generalisation: How much this improves the error?

What can we learn after one GD step ?

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (y_i - f(x_i; a^0, W^0))^2$$

Two flavours of results:

1. Weak learnability: How much W^1 correlates with $f_\star$?

After 1 step, can learn at best a non-linear function of a direction with $n = \Theta(d)$ samples

$$f_\star(x) \approx g(\langle \theta_\star, x \rangle) + \text{noise}$$

2. Generalisation: How much this improves the error?

What can we learn after one GD step ?

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (y_i - f(x_i; a^0, W^0))^2$$

Two flavours of results:

1. Weak learnability: How much W^1 correlates with $f_\star$?

After 1 step, can learn at best a non-linear function of a direction with $n = \Theta(d)$ samples

$$f_\star(x) \approx g(\langle \theta_\star, x \rangle) + \text{noise}$$

2. Generalisation: How much this improves the error?

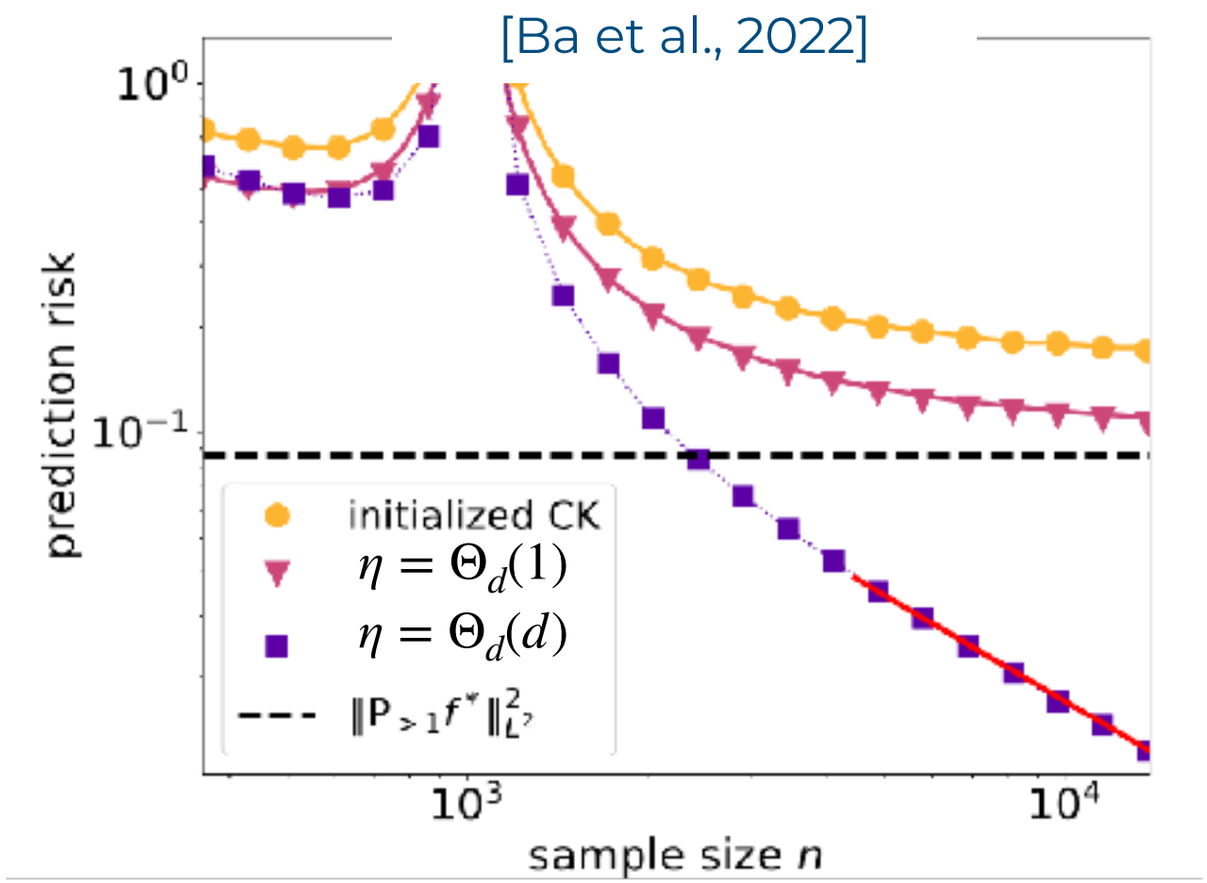
One step of GD

$$\hat{a}_\lambda(X,y) = \underset{a\in\mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2n}\sum_{i=1}^{n} (g(\langle\theta_\star, x_i\rangle) - \langle a, \sigma(W^1 x_i))^2 + \lambda\,||a||_2^2$$

One step of GD

$$\hat{a}_\lambda(X,y) = \underset{a\in\mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2n}\sum_{i=1}^{n} (g(\langle\theta_\star, x_i\rangle) - \langle a, \sigma(W^1 x_i)\rangle^2 + \lambda\,||a||_2^2$$

[Ba et al., 2022]



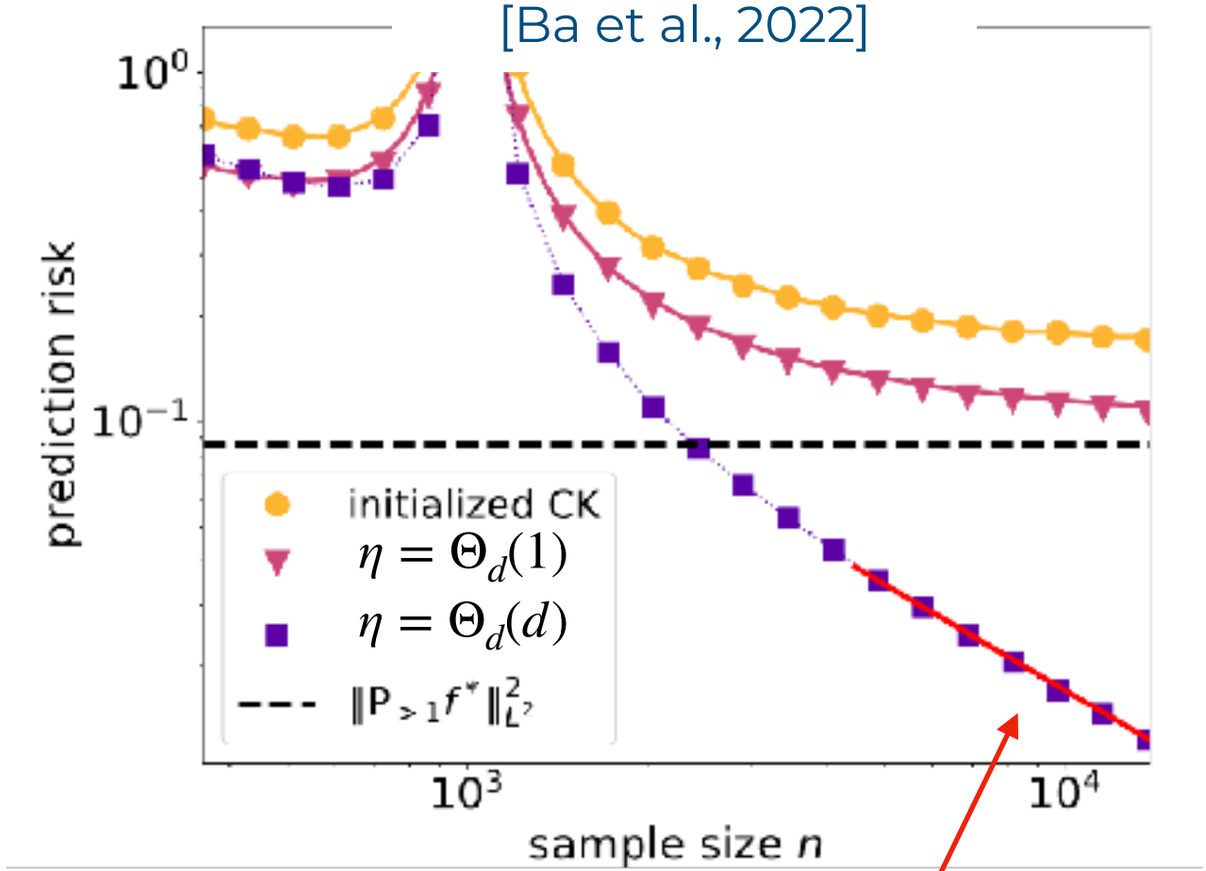
- For $n, p = \Theta(d)$ and $\eta = \Theta(1)$, no! GEP still valid.

- $\eta = \Theta_d(d)$ sufficient to learn more.

One step of GD

$$\hat{a}_\lambda(X,y) = \underset{a\in\mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2n}\sum_{i=1}^{n} (g(\langle\theta_\star, x_i\rangle) - \langle a, \sigma(W^1 x_i)\rangle^2 + \lambda\,||a||_2^2$$

[Ba et al., 2022]



- For $n, p = \Theta(d)$ and $\eta = \Theta(1)$, no! GEP still valid.

- $\eta = \Theta_d(d)$ sufficient to learn more.

Can we get that curve?

Gradient after 1 step

After a single gradient step with $n, p, \eta = \Theta(d)$:

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (g(\langle \theta_\star, x_i \rangle) - f(x_i; a^0, W^0))^2$$

We can decompose:

$$W^1 = W^0 + \breve{u}\breve{v} + \Delta$$

$$\breve{u} = \eta \mu_1 a^0 \in \mathbb{R}^p \qquad \breve{v} = \frac{1}{n_B} \sum_{i=1}^{n_B} \breve{\sigma}(W^0 x_i) g(\langle \theta_\star, x_i \rangle) x_i \in \mathbb{R}^d$$

$$\breve{\sigma}(z) = \sigma(z) - \mu_1$$
$$\mu_1 = \mathbb{E}[\sigma(z)z]$$

Taking $a^0 = 1_p$, after some massage...

Gradient after 1 step

After a single gradient step with $n, p, \eta = \Theta(d)$:

$$W^1 = W^0 - \frac{\eta}{2n} \sum_{i=1}^{n_B} \nabla_w (g(\langle \theta_\star, x_i \rangle) - f(x_i; a^0, W^0))^2$$

We can decompose:

$$W^1 \approx W + ruv$$

$$r = \frac{\eta}{d}\frac{p}{d}\mu_1 \sqrt{\frac{d}{n_B}\mu_2^\star + \mu_1^{\star 2}} \qquad c = 1 + \frac{\eta^2 d}{n_B p^2}\mu_1^2 \breve{\mu}_1^2 \mu_2^\star \qquad \langle v, \theta_\star \rangle = \frac{\mu_1^\star}{\sqrt{\frac{d}{n_B}\mu_2^\star + \mu_1^{\star 2}}}$$

$$w_k \in \mathbb{S}^{d-1}(\sqrt{c}) \qquad u \in \mathbb{S}^{d-1}(\sqrt{p}) \qquad v \in \mathbb{S}^{d-1}$$

$$\mu_1 = \mathbb{E}[\sigma(z)z] \qquad \mu_2 = \mathbb{E}[\sigma(z)^2] \qquad \breve{\mu}_1^2 = \mathbb{E}[(\sigma(z)z - \mu_1)^2]$$

Why this is hard?

Challenge: Characterise the properties of random matrices of the type

$$\Phi = \sigma\left(X(W^\top + vu^\top)\right)$$

Spiked Random Features model

$w_k \in \mathbb{S}^{d-1}(\sqrt{c})$ $x \sim \mathcal{N}(0, I_d/d)$ $v \in \mathbb{S}^{d-1}$ $u \in \mathbb{S}^{d-1}(\sqrt{p})$

Challenge:

Conditional GEP

Recall that for the standard RF model

Gaussian Equivalence Theorem (GET)

$$\sigma\left(\langle w^0, x\rangle\right) \approx \mu_0 + \mu_1\langle w^0, x\rangle + \mu_\star \xi$$

[Goldt et al. 19; Mei, Montanari '19; Hu & Lu '20]

Conditional GEP

Recall that for the standard RF model

Gaussian Equivalence Theorem (GET)

$$\sigma\left(\langle w^0, x\rangle\right) \approx \mu_0 + \mu_1\langle w^0, x\rangle + \mu_\star \xi$$

[Goldt et al. 19; Mei, Montanari '19; Hu & Lu '20]

We can show that for a sRF model with $a^0 = 1_p$:

cGET [Dandi, Krzakala, **BL**, Pesce, Stephan '23]

$$\sigma\left(\langle w^1, x\rangle\right) \approx \mu_0(\langle v, x\rangle) + \mu_1(\kappa)\langle w^0, x^\perp\rangle + \mu_\star(\kappa)\xi$$

$$\kappa = \langle v, x\rangle \qquad x = \kappa\theta_\star + x^\perp$$

Conditional GEP

Recall that for the standard RF model

Gaussian Equivalence Theorem (GET)

$$\sigma\left(\langle w^0, x\rangle\right) \approx \mu_0 + \mu_1\langle w^0, x\rangle + \mu_\star \xi$$

[Goldt et al. 19; Mei, Montanari '19; Hu & Lu '20]

We can show that for a sRF model with $a^0 = 1_p$:

cGET [Dandi, Krzakala, **BL**, Pesce, Stephan '23]

$$\sigma\left(\langle w^1, x\rangle\right) \approx \mu_0(\langle v, x\rangle) + \mu_1(\kappa)\langle w^0, x^\perp\rangle + \mu_\star(\kappa)\xi$$

$$\kappa = \langle v, x\rangle \qquad x = \kappa\theta_\star + x^\perp$$

Examples: $\sigma(z) = \text{sign}$

$$\mu_0(\kappa) = \text{erf}\left(\frac{\kappa}{\sqrt{2}}\right) \quad \mu_1(\kappa) = \sqrt{\frac{2}{\pi}}e^{-\frac{1}{2}\kappa^2}$$

$$\mu_2(\kappa) = 1 - \mu_0(\kappa)^2 - \mu_1(\kappa)^2$$

Main result

Together, this allow us to characterise the risk:

$$R(\hat{a}_\lambda) = \mathbb{E}[(g(\langle \theta_\star, x \rangle) - \langle \hat{a}_\lambda, \sigma(W^1 x_i))^2]$$

Where:

$$\hat{a}_\lambda(X, y) = \underset{a}{\operatorname{argmin}} \frac{1}{2n} \sum_{i=1}^{n} (g(\langle \theta_\star, x_i \rangle - \langle a, \sigma(W^1 x_i))^2 + \lambda ||a||_2^2$$

Main result

Together, this allow us to characterise the risk:

$$R(\hat{a}_\lambda) = \mathbb{E}[(g(\langle \theta_\star, x \rangle) - \langle \hat{a}_\lambda, \sigma(W^1 x_i))^2]$$

Where:

$$\hat{a}_\lambda(X, y) = \underset{a}{\operatorname{argmin}} \frac{1}{2n} \sum_{i=1}^{n} (g(\langle \theta_\star, x_i \rangle - \langle a, \sigma(W^1 x_i))^2 + \lambda ||a||_2^2$$

More precisely, for $a^0 = 1_p$ in the limit $d \to \infty$ with $n, p, \eta = \Theta(d)$:

$$R(\hat{a}_\lambda) = \mathbb{E}_{\kappa,z} \left[\left(g\left(\gamma\kappa + \sqrt{1-\gamma^2} z\right) - \mu_0(\kappa) m - \mu_1(\kappa)\kappa\zeta - \frac{\mu_1(\kappa)\psi}{\sqrt{\rho}} z \right)^2 + \mu_1(\kappa)^2 q_1 + \mu_2(\kappa)^2 q_2 - \frac{\mu_1(\kappa)^2 \psi^2}{\rho} \right]$$

$$m = \frac{1^\top \hat{a}_\lambda}{\sqrt{p}} \qquad q_1 = \frac{\langle W^\top \hat{a}_\lambda, \Pi^\perp W^\top \hat{a}_\lambda \rangle}{p} \qquad q_2 = \frac{||\hat{a}_\lambda||_2^2}{p} \qquad \zeta = \frac{\langle \hat{a}_\lambda, Wv \rangle}{\sqrt{dp}}$$

Exact asymptotics ($a^0 = 1_p$)

$$
\begin{cases}
V_1 = \int \dfrac{d\nu(\varrho,\tau,\pi)\varrho}{\lambda + \hat{V}_1\varrho + \hat{V}_2} \\
V_2 = \int \dfrac{d\nu(\varrho,\tau,\pi)}{\lambda + \hat{V}_1\varrho + \hat{V}_2} \\
m = \dfrac{\mathbb{E}_{\kappa,y}\left[\frac{\mu_0(\kappa)(\sigma_\star(\kappa,y) - \mu_1(\kappa)\kappa\zeta)}{1+V(\kappa)}\right]}{\mathbb{E}_{\kappa}\left[\frac{\mu_0(\kappa)^2}{1+V(\kappa)}\right]} \\
\zeta = \hat{\zeta}\sqrt{\beta}\int d\nu(\varrho,\tau,\pi)\varrho\tau^2 \dfrac{1}{\lambda + \hat{V}_1\varrho + \hat{V}_2} + \beta^{\frac{3}{2}}\hat{\zeta}\hat{V}_1 \dfrac{I(\hat{V}_1,\hat{V}_2)^2}{1-\beta\hat{V}_1 I(\hat{V}_1,\hat{V}_2)} \\
\psi = \hat{\psi}\sqrt{\beta}\int \dfrac{d\nu(\varrho,\tau,\pi)\varrho\pi^2}{\lambda + \hat{V}_1\varrho + \hat{V}_2}
\end{cases}
$$

$$
\begin{cases}
q_1 = \int d\nu(\varrho,\tau,\pi)\varrho \dfrac{\left(\hat{q}_1\varrho + \hat{q}_2 + \hat{\zeta}^2\varrho\tau^2 + \hat{\psi}^2\varrho\pi^2\right)}{\left(\lambda + \hat{V}_1\varrho + \hat{V}_2\right)^2} - \beta\hat{\zeta}^2 \dfrac{I(\hat{V}_1,\hat{V}_2)^2}{\left(1-\beta\hat{V}_1 I(\hat{V}_1,\hat{V}_2)\right)^2} \\
\qquad -\hat{\zeta}^2 \dfrac{\int \frac{\tau^2\varrho^2 d\nu(\varrho,\tau,\pi)}{(\lambda+\hat{V}_1\varrho+\hat{V}_2)^2}\left[\left(1-\beta\hat{V}_1 I(\hat{V}_1,\hat{V}_2)\right)^2 - 1\right]}{\left(1-\beta\hat{V}_1 I(\hat{V}_1,\hat{V}_2)\right)^2} \\
q_2 = \int \dfrac{\left(\hat{q}_1\varrho + \hat{q}_2 + \hat{\zeta}^2\varrho\tau^2 + \hat{\psi}^2\varrho\pi^2\right)d\nu(\varrho,\tau,\pi)}{\left(\lambda + \hat{V}_1\varrho + \hat{V}_2\right)^2} \\
\qquad -\hat{\zeta}^2\int \dfrac{\tau^2\varrho\, d\nu(\varrho,\tau,\pi)}{(\lambda+\hat{V}_1\varrho+\hat{V}_2)^2}\left[1 - \dfrac{1}{\left(1-\beta\hat{V}_1 I(\hat{V}_1,\hat{V}_2)\right)^2}\right]
\end{cases}
$$

$$
\begin{cases}
\hat{V}_1 = \frac{\alpha}{\beta}\mathbb{E}_\kappa \frac{\rho\mu_1(\kappa)^2}{1+V(\kappa)} \\
\hat{V}_2 = \frac{\alpha}{\beta}\mathbb{E}_\kappa \frac{\rho\mu_2(\kappa)^2}{1+V(\kappa)} \\
\hat{\zeta} = \frac{\alpha}{\sqrt{\beta}}\mathbb{E}_{\kappa,y}\kappa\mu_1(\kappa)\frac{b(\kappa,y)}{1+V(\kappa)} \\
\hat{\psi} = \frac{\alpha}{\sqrt{\beta}}\mathbb{E}_{\kappa,y}\frac{y\mu_1(\kappa)b(\kappa,y) + \psi\mu_1(\kappa)^2}{1+V(\kappa)}
\end{cases}
$$

$$
\begin{cases}
\hat{q}_1 = \frac{\alpha}{\beta}\mathbb{E}_{\kappa,y}\mu_1(\kappa)^2 \frac{b(\kappa,y)^2 + \rho q(\kappa) - \mu_1(\kappa)^2\psi^2}{\left(1+V(\kappa)\right)^2} \\
\hat{q}_2 = \frac{\alpha}{\beta}\mathbb{E}_{\kappa,y}\mu_2(\kappa)^2 \frac{b(\kappa,y)^2 + \rho q(\kappa) - \mu_1(\kappa)^2\psi^2}{\left(1+V(\kappa)\right)^2}
\end{cases}
$$

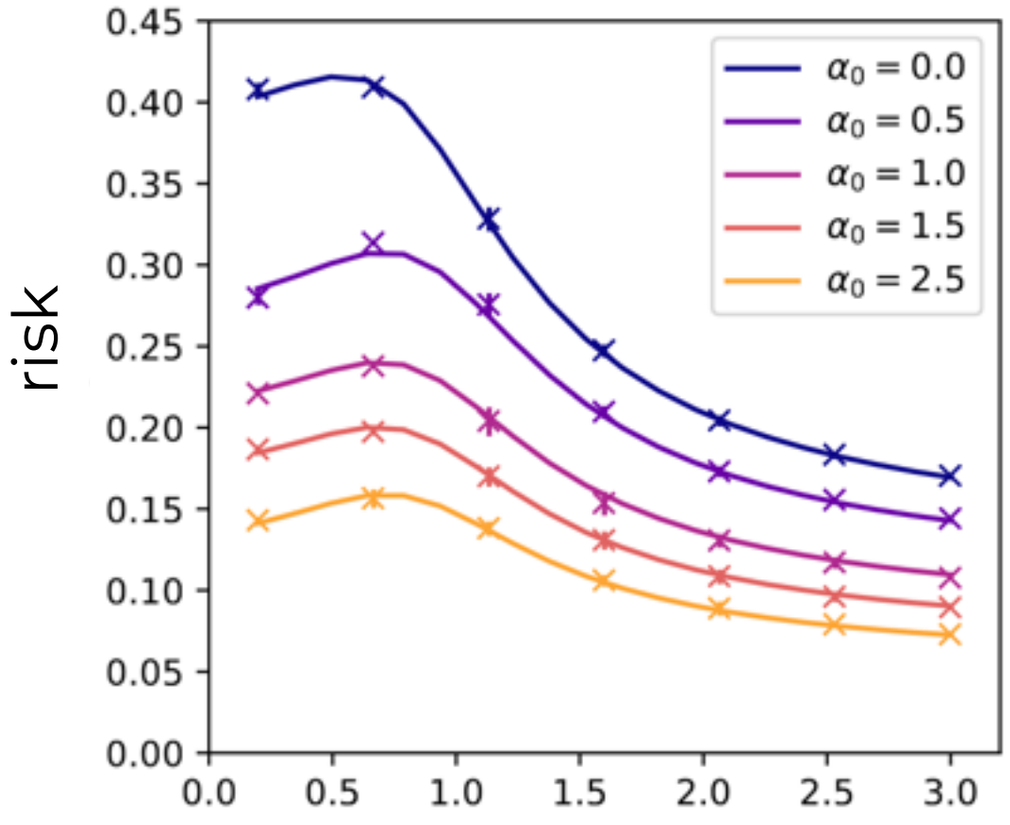
$$
\alpha_0 = n_B/d \qquad \beta = p/d
$$
$$
\alpha = n/d \qquad \tilde{\eta} = \eta/d
$$
$$
\kappa = \langle v, x\rangle \qquad \rho = 1-\gamma^2
$$
$$
\gamma = \langle v, \theta_\star\rangle
$$

$$
W = \sum_{i=1}^{\min(p,d)} \lambda_i e_i f_i^\top \qquad \Pi^\perp = I_d - vv^\top
$$
$$
\nu(\varrho,\tau,\pi) = \frac{1}{p}\sum_{i=1}^{\min(p,d)} \delta\left(\lambda_i - \varrho\right)\delta\left(f_i^\top v - \tau\right)\delta\left(f_i^\top \Pi^\perp\vec{\theta} - \pi\right)
$$

Batch size

$\tilde{\eta} = 1$ $\lambda = 10^{-2}$

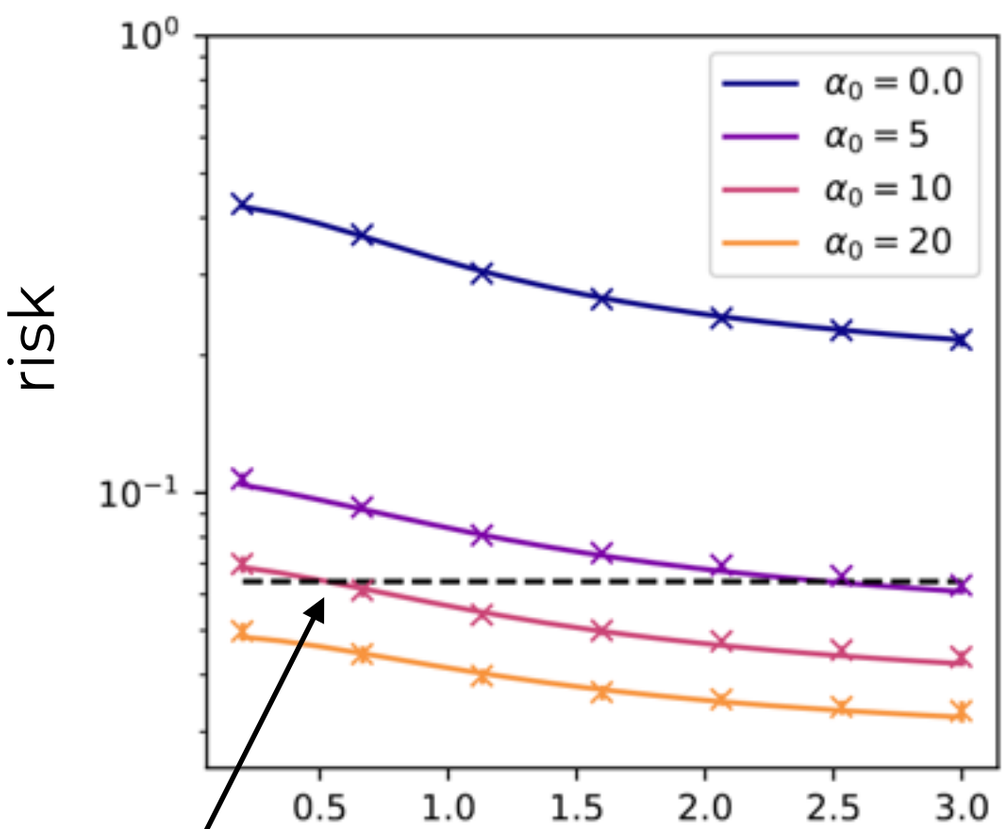
$\sigma = g = \tanh$



$\alpha = n/d$

$\tilde{\eta} = 3$ $\lambda = 0.1$

$\sigma = \tanh$ $g = \text{sign}$

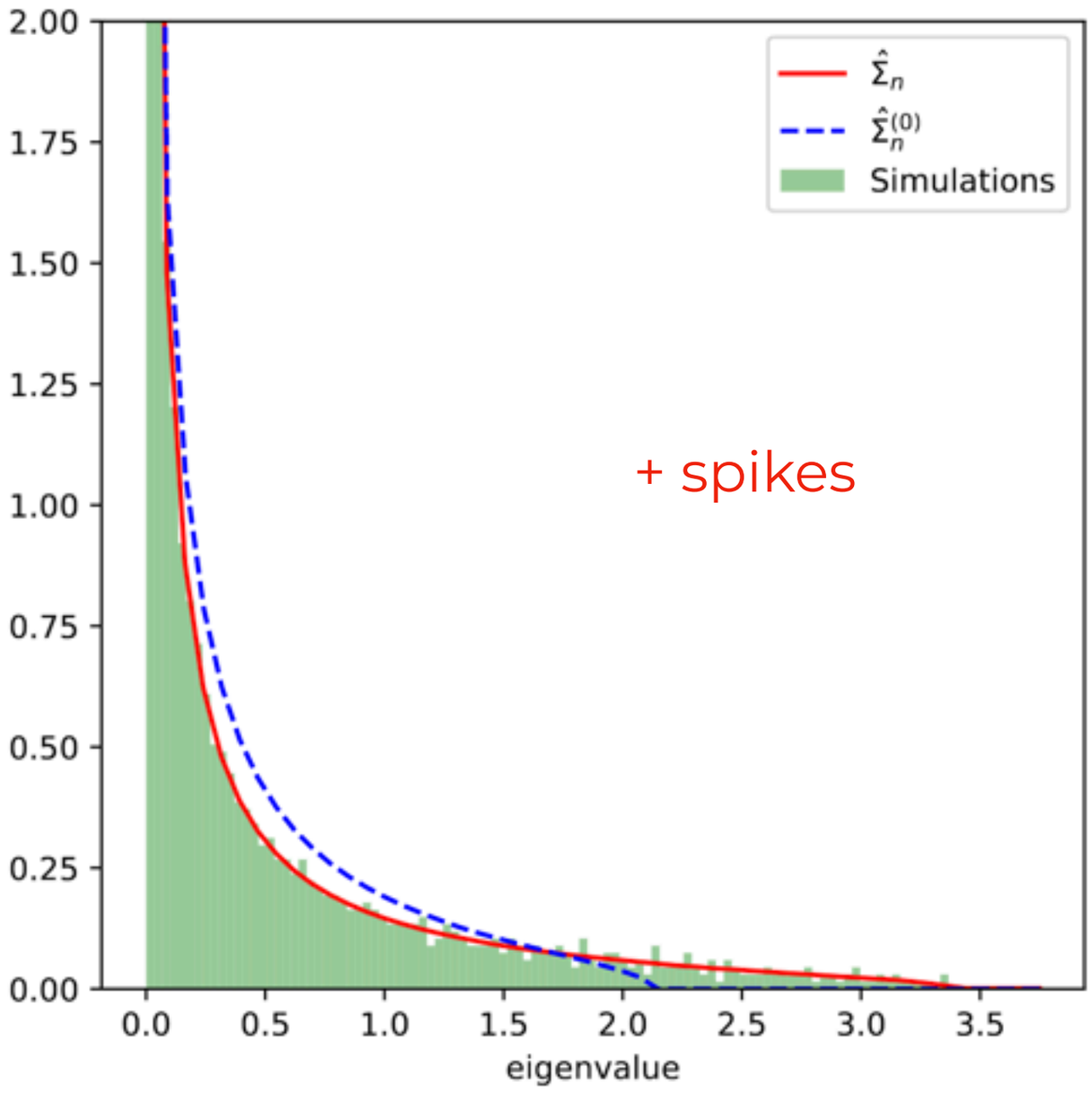


$\alpha = n/d$

Best linear predictor

$||P_{\kappa \leq 1} f_\star||^2$

Spectral properties



Risk bounds

Recall that. Noting that this is monotonic in $\alpha_0 = n_B/d$:

$$R(\hat{a}_\lambda) = \mathbb{E}_{\kappa,z}\left[\left(g\left(\gamma\kappa + \sqrt{1-\gamma^2}z\right) - \mu_0(\kappa)m - \mu_1(\kappa)\kappa\zeta - \frac{\mu_1(\kappa)\psi}{\sqrt{\rho}}z\right)^2 + \mu_1(\kappa)^2 q_1 + \mu_2(\kappa)^2 q_2 - \frac{\mu_1(\kappa)^2\psi^2}{\rho}\right]$$

Risk bounds

Recall that. Noting that this is monotonic in $\alpha_0 = n_B/d$:

$$\inf_{\lambda \geq 0} R(\alpha, \lambda, \tilde{\eta}, \beta) \leq \inf_{b_1} \mathbb{E}[(g(\kappa) - b_1 \mu_0(\kappa))^2]$$

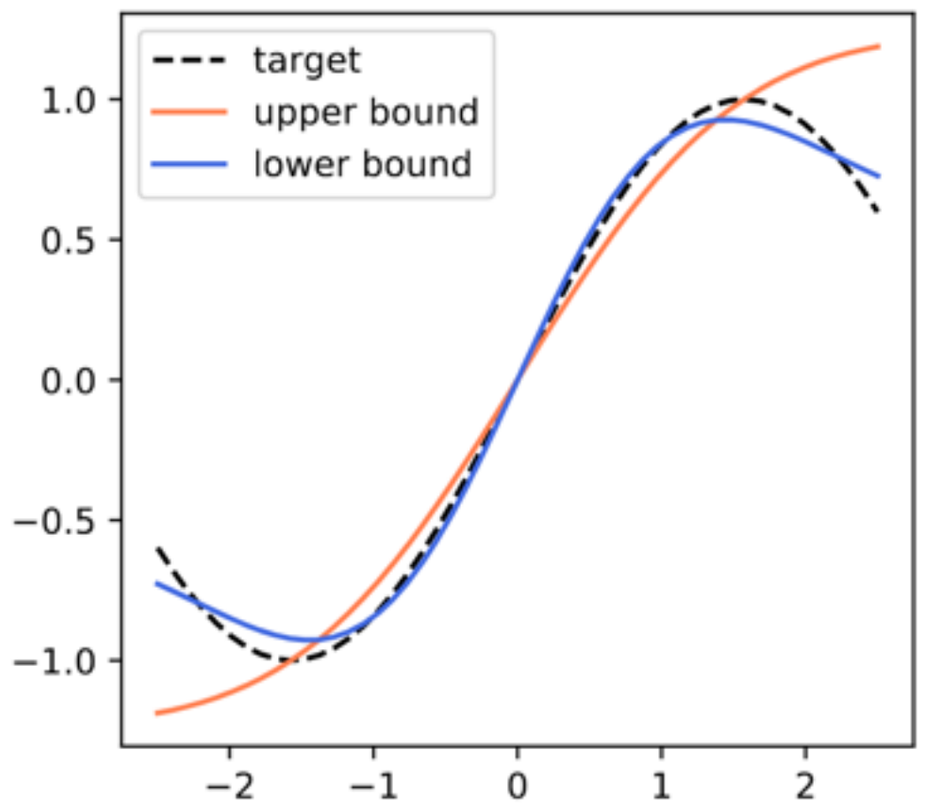
$$\inf_{\lambda \geq 0} R(\alpha, \lambda, \tilde{\eta}, \beta) \geq \inf_{b_1, b_2} \mathbb{E}[(g(\kappa) - b_1 \mu_0(\kappa) - b_2 \mu_1(\kappa)\kappa)^2]$$

$c = \gamma = 1$

$r = 0.9$

$g = \sin$

$\sigma = \tanh$



Risk bounds

Recall that. Noting that this is monotonic in $\alpha_0 = n_B/d$:

$$\inf_{\lambda \geq 0} R(\alpha, \lambda, \tilde{\eta}, \beta) \leq \inf_{b_1} \mathbb{E}[(g(\kappa) - b_1 \mu_0(\kappa))^2]$$

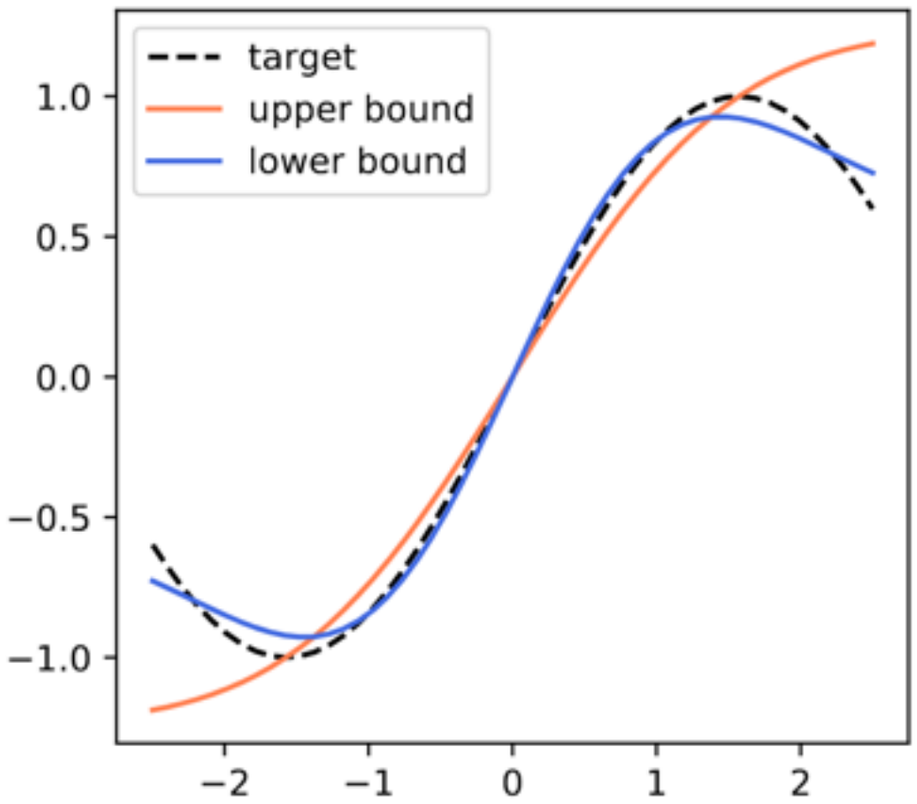
$$\inf_{\lambda \geq 0} R(\alpha, \lambda, \tilde{\eta}, \beta) \geq \inf_{b_1, b_2} \mathbb{E}[(g(\kappa) - b_1 \mu_0(\kappa) - b_2 \mu_1(\kappa)\kappa)^2]$$

$c = \gamma = 1$

$r = 0.9$

$g = \sin$

$\sigma = \tanh$



n.b.:

1. $L_2(\mathcal{N})$ distance between g and $\text{span}(\mu_0, \mu_1')$

2. Can make tighter by optimising over $\tilde{\eta}$

A note on initialisation

So far, assumed $a^0 = 1_p$. But can be generalised to finite support $a^0 \in V$.

$$\sigma(W^1 x) \asymp \begin{bmatrix} \mu_0(u_1\kappa) \\ \vdots \\ \mu_0(u_p\kappa) \end{bmatrix} + \begin{bmatrix} \mu_1(u_1\kappa) \\ \vdots \\ \mu_1(u_p\kappa) \end{bmatrix} \odot Wx + \begin{bmatrix} \mu_2(u_1\kappa) \\ \vdots \\ \mu_2(u_p\kappa) \end{bmatrix} \odot \xi$$

$$u \in V^p \qquad\qquad \xi \sim \mathcal{N}(0, I_p)$$

A note on initialisation

So far, assumed $a^0 = 1_p$. But can be generalised to finite support $a^0 \in V$.

$$\sigma(W^1 x) \asymp \begin{bmatrix} \mu_0(u_1\kappa) \\ \vdots \\ \mu_0(u_p\kappa) \end{bmatrix} + \begin{bmatrix} \mu_1(u_1\kappa) \\ \vdots \\ \mu_1(u_p\kappa) \end{bmatrix} \odot Wx + \begin{bmatrix} \mu_2(u_1\kappa) \\ \vdots \\ \mu_2(u_p\kappa) \end{bmatrix} \odot \xi$$

$$u \in V^p \qquad \xi \sim \mathcal{N}(0, I_p)$$

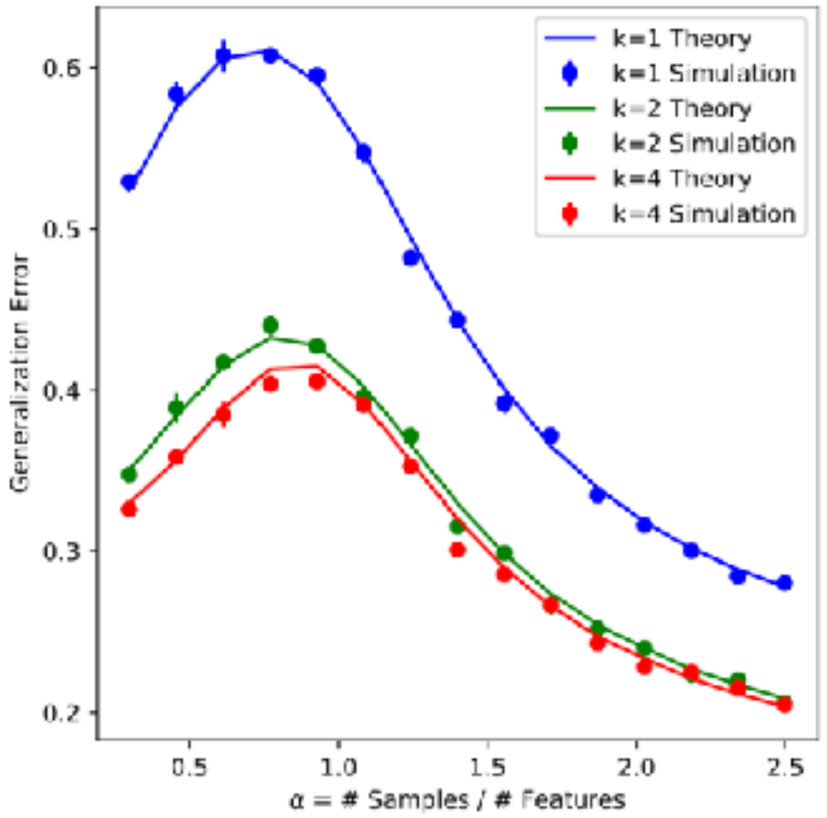
This now spans a richer functional basis:

$$\{\mu_0(\omega\cdot), \mu_1'(\omega\cdot)\}_{\omega \in V}$$

For instance, in the limit $\lambda, \alpha_0, \tilde{\eta} \to \infty$:

$$\sigma(W^1 x)_k \asymp \mu_0(u_k\kappa)$$

Single neuron with random weights.



Proof idea

Main ideas

SGD step ⟶ sRF model ⟶ cGET

$$\varphi_i = \sigma(W_1 x_i) \quad \approx \quad \sigma(\tilde{W} x_i + \langle v, x_i \rangle u^\top) \quad \approx \quad \mu_0(\kappa_i u) + \mu_1(\kappa_i u)\tilde{W} x_i^\perp + \mu_\star(\kappa_i u)\xi_i$$

Main ideas

SGD step ⟶ sRF model ⟶ cGET

$$\varphi_i = \sigma(W_1 x_i) \quad \approx \quad \sigma(\tilde{W} x_i + \langle v, x_i \rangle u^\top) \quad \approx \quad \mu_0(\kappa_i u) + \mu_1(\kappa_i u)\tilde{W} x_i^\perp + \mu_\star(\kappa_i u)\xi_i$$

2 stages of deterministic equivalent: over X and $\tilde{W}$
(leave-one-out + Burkholder)

Main challenges:

- For $u_j \in \{\zeta_1, \ldots, \zeta_k\}$, with prob. $\pi_j = p_j/p$, need to handle k spikes separately.
- For bulk, need deterministic equivalent for block-structured Wishart matrices

$$M = (C_e \odot \tilde{W}\tilde{W}^\top + D_e)^{-1}$$

$$\sum_{j=1}^{k} p_j = p$$

$$C_e = \begin{bmatrix} C_{11}1_{p_1\times p_1} & C_{12}1_{p_1\times p_2} & \cdots & C_{1k}1_{p_1\times p_k} \\ C_{21}1_{p_2\times p_1} & C_{22}1_{p_2\times p_2} & \cdots & C_{2k}1_{p_2\times p_k} \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix} \in \mathbb{R}^{k\times k}$$

$$D_e = \begin{bmatrix} D_{11}I_{p_1\times p_1} & 0 & \cdots & 0 \\ 0 & D_{22}I_{p_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix} \in \mathbb{R}^{k\times k}$$

Conclusion

In proportional asymptotics,
kernels can learn at best a linear approximation

With one gradient step, 2LNN learn
do better than kernels along
one (and only one) direction

We can provide a sharp asymptotic description
on what is learned

Collaborators in these works

L. Zdeborová
(EPFL)

F. Krzakala
(EPFL)

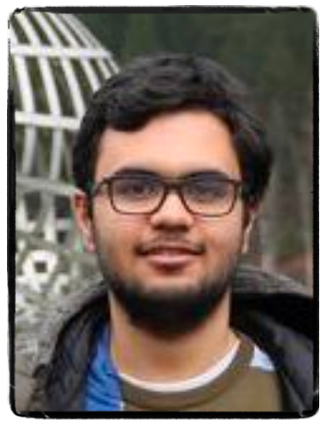
L. Stephane
(EPFL)

G. Reeves
(Duke U.)

Y.M. Lu
(Harvard U.)

H. Cui
(EPFL)

L Pesce
(EPFL)

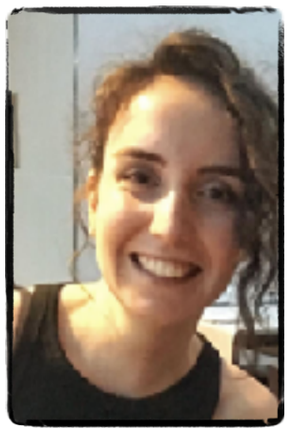


Y. Dandi
(EPFL)

T. Misiakiewicz
(Yale)

L. Defilippis
(DI-ENS)

S. Goldt
(SISSA)



F. Gerace
(SISSA)

M. Mézard
(Bocconi U.)

Thank you!