LEARNING SPARSE FEATURES CAN LEAD TO OVERFITTING IN NEURAL NETWORKS

Leonardo Petrini*◆, Francesco Cagnetta*◆, Eric Vanden-Eijnden◉, Matthieu Wyart◆

leonardo.petrini@epfl.ch @leopetrini_

francesco.cagnetta@epfl.ch

*Equal Contribution

◆Institute of Physics, EPFL

◉Courant Institute of Mathematical Sciences, NYU

IN SHORT

1. Common idea: **deep nets** success is in their ability to **learn** data **features**;
2. Is **learning features** actually good for **performance**? In image tasks:
   👍 ***Yes*** for **convolutional nets** (CNNs) but 👎 ***No*** for **full conn. nets** (FCNs)

   We propose an explanation for this puzzle:
3. **Feature learning** can perform worse than **lazy training** as it leads to a **sparser** neural representation.
4. **Sparse representations** are detrimental when the target function is **constant / smooth** along some input-space directions.
5. We illustrate this is *(i)* **regression of Gaussian random functions on the *d*-sphere** and *(ii)* benchmark **image datasets**.

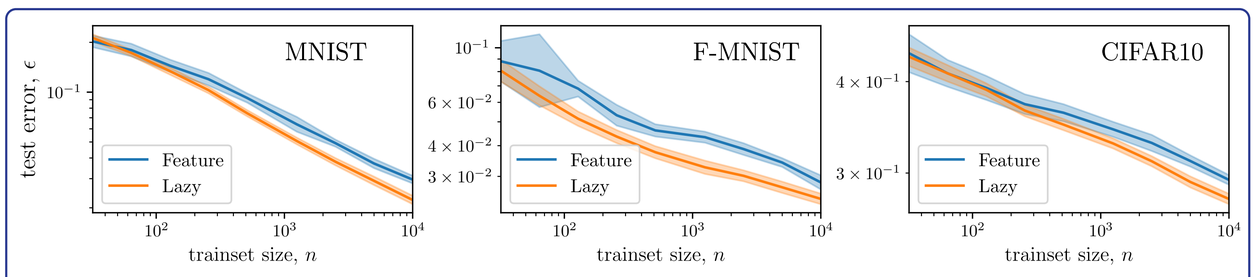


Figure 1: Test error vs. training-set size of infinite-width FCNs trained on image classification

RANDOM FUNCTIONS ON THE SPHERE

Task: supervised learning with n training points $\{\boldsymbol{x}_i\}_{i=1}^n$ uniform on the sphere $\mathbb{S}^{d-1}$ and target f^* Gaussian random process with controlled power spectrum decay:

$$f^*(\boldsymbol{x}) = \sum_{k\geq 0}\sum_{\ell=1}^{\mathcal{N}_{k,d}} f^*_{k,\ell} Y_{k,\ell}(\boldsymbol{x}) \quad \text{with} \quad \mathbb{E}\left[f^*_{k,\ell}\right]=0, \quad \mathbb{E}\left[f^*_{k,\ell}f^*_{k',\ell'}\right] = c_k \delta_{k,k'}\delta_{\ell,\ell'}$$

f^*: target function; $f^*_{k,\ell}$: coefficients; $Y_{k,\ell}$: spherical harmonics

$$c_k \sim k^{-2\nu_t-(d-1)} \text{ for } k \gg 1$$

power spectrum decay

Figure 2: data samples

which determines the target smoothness in real space

$$\mathbb{E}\left[|f^*(\boldsymbol{x})-f^*(\boldsymbol{y})|^2\right] = O\left(|\boldsymbol{x}-\boldsymbol{y}|^{2\nu_t}\right) = O\left((1-\boldsymbol{x}\cdot\boldsymbol{y})^{\nu_t}\right) \quad \text{as} \quad \boldsymbol{x}\to\boldsymbol{y}.$$

Smoothness

NEURAL NETS TRAINING REGIMES

We consider a one-hidden-layer ReLU net of width H:

$$f_H^{\xi}(\boldsymbol{x}) = \frac{1}{H^{1-\xi/2}}\sum_{h=1}^{H}\left(w_h\sigma(\boldsymbol{\theta}_h\cdot\boldsymbol{x}) - \xi w_h^0\sigma(\boldsymbol{\theta}_h^0\cdot\boldsymbol{x})\right),$$

f_H^{ξ}: network predictor; w_h: weights (optimized params); $\boldsymbol{\theta}_h$: features; $w_h^0, \boldsymbol{\theta}_h^0$: initialization params

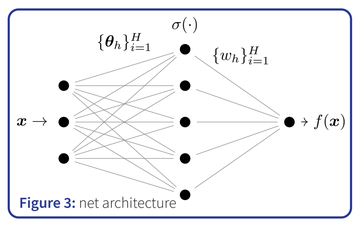


Figure 3: net architecture

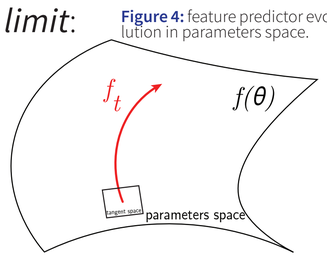
where ξ controls the training regime, i.e. it is zero in the *feature regime* and one in the *lazy regime* such that we get well-defined $H\to\infty$ limits:

• ***Feature regime.*** In this regime we get the so-called *mean-field limit*:

$$\lim_{H\to\infty} f_H^{\xi=0}(\boldsymbol{x}) = \int_{\mathbb{S}^{d-1}} \sigma(\boldsymbol{\theta},\boldsymbol{x})\, d\gamma(\boldsymbol{\theta}) \qquad \text{a.e. on } \mathbb{S}^{d-1}.$$

Radon measure

Figure 4: feature predictor evolution in parameters space.



and the optimal γ having minimal norm is determined by

$$\gamma^* = \arg\min_{\gamma}\int_{\mathbb{S}^{d-1}}|d\gamma(\boldsymbol{\theta})| \quad \text{subject to} \quad \int_{\mathbb{S}^{d-1}}\sigma(\boldsymbol{\theta}\cdot\boldsymbol{x}_i)d\gamma(\boldsymbol{\theta}) = f^*(\boldsymbol{x}_i) \quad \forall i=1,\dots,n.$$

Notice that this is equivalent to Lasso regression and hence the solution is unique and sparse (i.e. supported on $n_A \leq n$ neurons) with probability 1. The predictor reads,

$$f^{\text{FEATURE}}(\boldsymbol{x}) = \sum_{i=1}^{n_A} w_i^*\sigma(\boldsymbol{\theta}_i^*\cdot\boldsymbol{x}).$$

• ***Lazy Regime.*** In this case we get the *Neural Tangent Kernel* (NTK) [1] limit where the neural network training correspond to kernel regression with the NTK and the predictor takes the form,

$$f^{\text{NTK}}(\boldsymbol{x}) = \sum_{i=1}^{n} g_i K^{\text{NTK}}(\boldsymbol{x}_i\cdot\boldsymbol{x}).$$

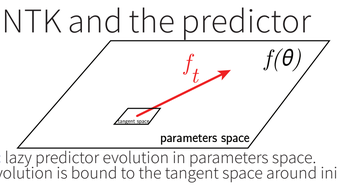


Figure 5: lazy predictor evolution in parameters space. Notice that evolution is bound to the tangent space around init.

GENERALIZATION ERROR ASYMPTOTICS

In the setting we introduced, we characterize the asymptotic decay of the generalization error with the number of training points averaged over realization of the target function:

$$\bar{\epsilon}(n) = \mathbb{E}_{f^*}\left[\int d\tau^{d-1}(\boldsymbol{x})\,\left(f^n(\boldsymbol{x})-f^*(\boldsymbol{x})\right)^2\right] = \mathcal{A}_d n^{-\beta} + o(n^{-\beta})$$

where the predictor $f^n(\boldsymbol{x})$, for both regimes, can be casted as a convolution on $\mathbb{S}^{d-1}$,

$$f^n(\boldsymbol{x}) = \sum_{j=1}^{O(n)} g_j\varphi(\boldsymbol{x}\cdot\boldsymbol{y}_j) := \int_{\mathbb{S}^{d-1}} g^n(\boldsymbol{y})\varphi(\boldsymbol{x}\cdot\boldsymbol{y})d\tau(\boldsymbol{y})$$

NOTATION

$d\tau$ uniform measure on the sphere

$g^n(\boldsymbol{x}) = \sum_j |\mathbb{S}^{d-1}| g_j \delta(\boldsymbol{x}-\boldsymbol{y}_j)$

which becomes a product in the basis of spherical harmonics: $f^n_{k,\ell} = g^n_{k,\ell}\varphi_k$.

The predictor for the test error decay β relies on the **spectral bias** ansatz: for the first n modes, the predictor $f^n_{k,\ell}$ coincides with the target function $f^*_{k,\ell}$ and the test error reads

$$\epsilon(n) \sim \sum_{k\geq k_c}^{\text{spectral bias}}\sum_{l=1}^{\mathcal{N}_{k,d}}\left(f^n_{k,l}-f^*_{k,l}\right)^2 \sim \sum_{k\geq k_c}\sum_{l=1}^{\mathcal{N}_{k,d}}\left(\underbrace{\langle(g^n_{k,l})^2\rangle\varphi_k^2}_{\text{predictor contribution}} + \underbrace{k^{-2\nu_t-(d-1)}}_{\text{target contribution}}\right).$$

from which one obtains

$$\beta^{\text{LAZY}} = \frac{\min\{2(d-1)+4\nu, 2\nu_t\}}{d-1} \text{ with } \nu = \begin{cases}1/2 \text{ for NTK},\\ 3/2 \text{ for RFK},\end{cases}$$

$$\beta^{\text{FEATURE}} = \frac{\min\{(d-1)+3, 2\nu_t\}}{d-1}.$$

Main Theoretical Result

NUMERICAL TESTS OF THE THEORY

We find good agreement between our theory and the training of a neural network with gradient descent and small regularization or the alpha-trick [2].

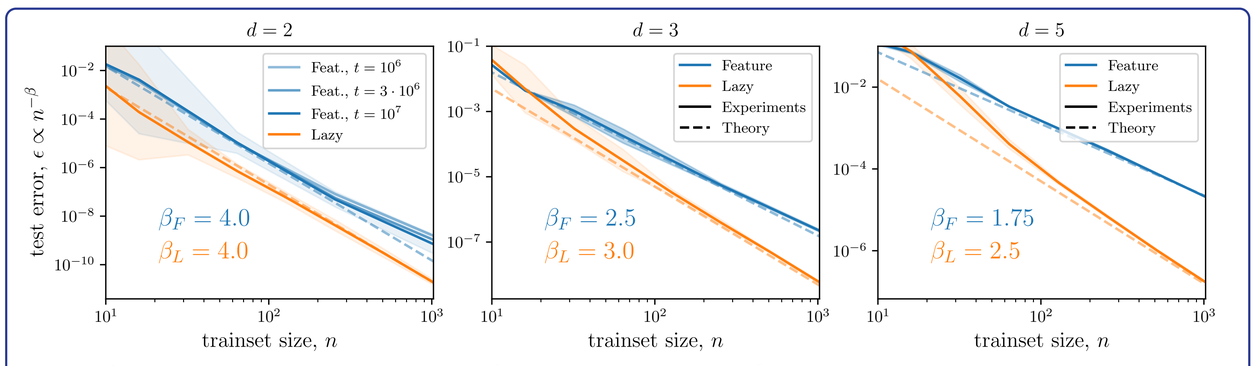


Figure 6: generalization error for a smooth target function, theory vs experiments.

OVERFITTING IN IMAGE DATASETS

Smooth target along non-linear input directions → **feature adaptation** leads to **sparse** representation = **overfitting**.

Does this picture hold for **images**? We argue yes because:

(i) the features distribution becomes **sparse**

(ii) the predictor of the **feature** regime is **less smooth** along directions for which the target should vary smoothly [3], i.e. **diffeomorphisms.**

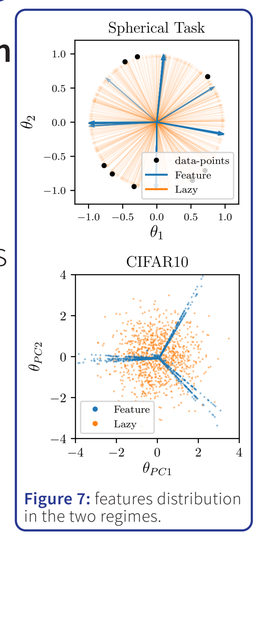


Figure 7: features distribution in the two regimes.

Figure 8: Measure of the smoothness of the predictor along diffeomorphims.

CONCLUSIONS AND FUTURE WORK

• **Learning features** is **detrimental** if task is **invariant / smooth** along transformations that are not captured by the network architecture;
• Our analysis relies on the **sparsity** of the **feature solution**;
• We provide test error decay predictions that we verify; this kind of results are scarce;
• May questions are still open. Understanding the **feature regime** in **modern architectures**: how do **CNNs perform well** in the feature regime? Does sparsity help?

Preprint: arXiv:XXX.XXXX.

Code: github.com/pcsl-epfl/regressionsphere

[1] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks, *NeurIPS '18.*

[2] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming, *NeurIPS '19.*

[3] Leonardo Petrini, Alessandro Favero, Mario Geiger, and Matthieu Wyart. Relative stability toward diffeomorphisms indicates performance in deep nets, *NeurIPS 21.*