On Lazy Training in Differentiable Programming

Lénaïc Chizat[3], Édouard Oyallon[4] and Francis Bach[1,2]
[1]INRIA, [2]ENS-PSL Paris, [3]CNRS, [4]Centrale-Supélec

Abstract

Context. Recent theory shows that training **large neural networks** amounts to doing regression with a positive-definite **kernel**.

Contributions. This phenomenon, that we call lazy training :

- is not intrinsically due to width but to a degenerate relative **scale**
  $\rightarrow$ depends on **early stopping**, **initialization** and **normalization**
- removes some benefits of depth and **hinders generalization**

Lazy Training

Setting. Adjust parameters of a differentiable model $h : \mathbb{R}^p \to \mathcal{F}$ by minimizing a loss $R : \mathcal{F} \to \mathbb{R}_+$ using gradient flow on the objective

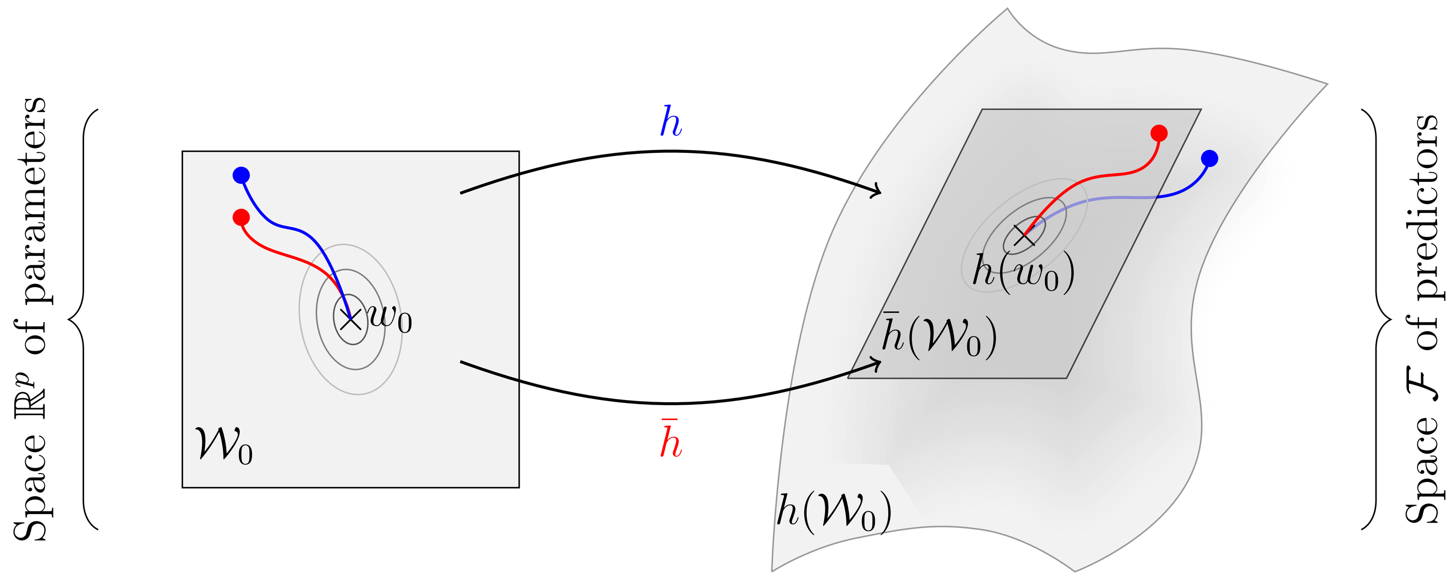
$$F(w) = R(h(w))$$

- $\mathcal{F}$ is a Hilbert space of predictors, R typically the empirical or population risk, h typically a neural network
- gradient flows approximate (stochastic, accelerated) gradient descent

Training paths. For initialization w_0 and stopping time T, let

- $(w(t))_{t\in[0,T]}$ be the *original* optimization path
- $(\bar{w}(t))_{t\in[0,T]}$ be the *tangent* optimization path, for the tangent model

$$\bar{h}(w) = h(w_0) + Dh(w_0)(w - w_0)$$



Lazy Training (definition)

When the *original* and *tangent* optimization paths are close

Consequences. Lazy training is a type of implicit bias for gradient descent that leads to strong guarantees:

- on optimization speed (theory of convex optimization)
- on generalization (theory of kernel regression)

When does lazy training occur?

Simple (sufficient) criterion. If over a training step it holds

$$\frac{\text{relative change of } F}{\text{relative change of } Dh} \ll 1$$

For the square loss $R(y) = \|y - y^\star\|^2$, leads to

$$\kappa_h(w_0) := \|h(w_0) - y^\star\| \frac{\|D^2 h(w_0)\|}{\|Dh(w_0)\|^2} \ll 1, \qquad (1)$$

Case 1: Rescaled models

For $\alpha > 0$, one has $\kappa_{\alpha h}(w_0) \lesssim \|h(w_0) - y^\star/\alpha\|$

Case 2: Homogeneous models

If $h(\lambda w) = \lambda^q h(w)$, one has $\kappa_h(\lambda w_0) \lesssim \|h(w_0) - y^\star/\lambda^q\|$
$\rightarrow$ lazy if $h(w_0)$ small and λ large

Case 3: Wide neural networks

If $h_m(w) = \alpha \sum_{i=1}^m \phi(\theta_i)$ where $w = (\theta_1, \dots, \theta_m)$ are i.i.d. and satisfy $\mathbb{E}\phi(\theta_i) = 0$ (two-layer neural network), then

$$\kappa_{h_m}(w_0) \lesssim m^{-1/2} + (\alpha m)^{-1}$$

$\rightarrow$ lazy if $\lim_{m\to\infty} \alpha m = \infty$ (e.g. $\alpha = 1/\sqrt{m}$)
$\rightarrow$ can be extended to deep networks (Jacot et al.)

Theoretical results

Finite horizon

If $h(w_0) = 0$, then training αh always becomes lazy as α grows.
If in particular $R(y) = \frac{1}{2}\|y - y^\star\|^2$, it holds

$$\frac{\|h(w(t)) - \bar{h}(\bar{w}(t))\|}{\|h(w_0) - y^\star\|} \lesssim (\underbrace{T \cdot \|Dh(w_0)\|^2}_{\approx \text{ nb. iterations}})^2 \cdot \kappa_h(w_0).$$

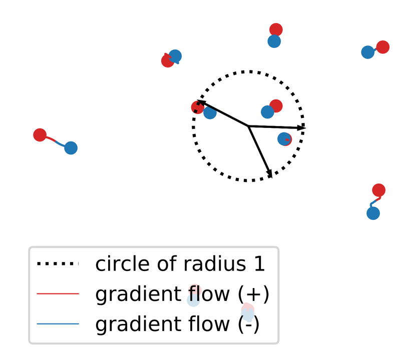
Infinite horizon

If $h(w_0) = 0$, and R is strongly convex, then training αh converges exponentially fast for large α, towards global (if over-parameterized) or local (if under-parameterized) minima.

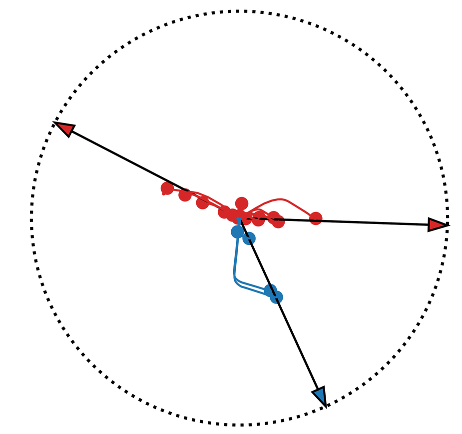
- see paper for precise statements.

Is lazy training desirable?

Synthetic experiments. Two-layer ReLU neural network in the teacher student-setting (square loss), initialized with variance τ.

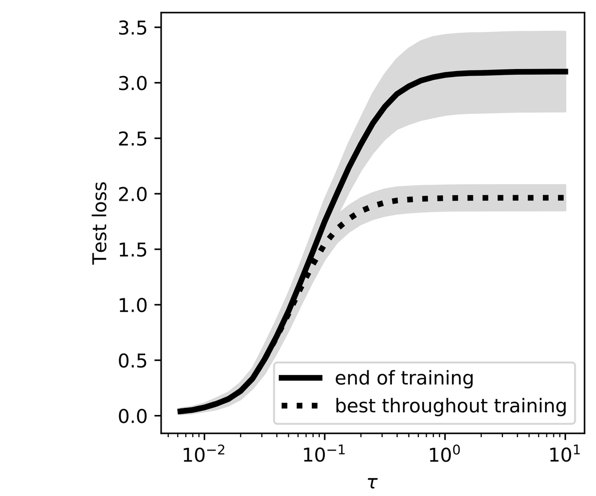


Lazy Training ($\tau = 0.1$)

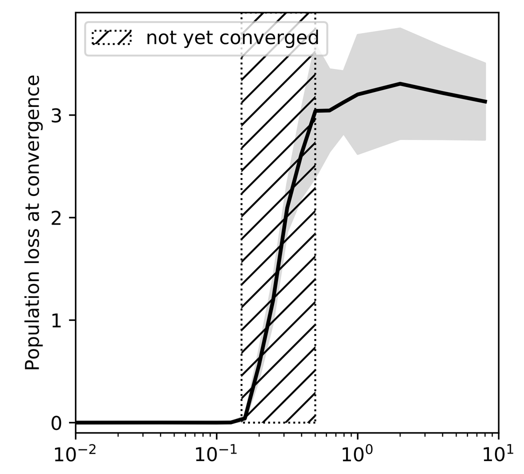


Non-Lazy Training ($\tau = 2$)

Trajectory of each "hidden" neuron during training (2-D input)



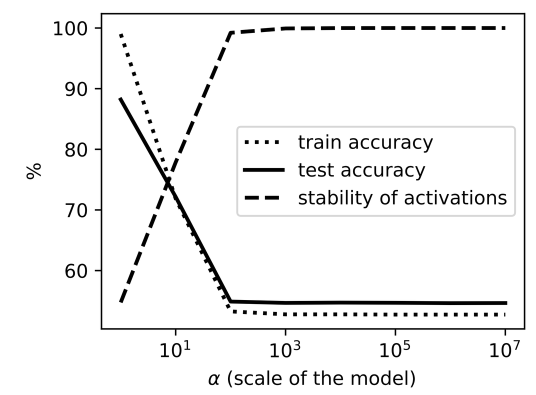
Over-parameterized
(GD on train loss until 0 loss)



Under-parameterized
(SGD on population loss)

Impact of laziness on performance (100-D input)

Image recognition task. Does lazy training explain deep learning?



Effect on laziness (VGG11 model)

Model	Train acc.	Test acc.
ResNet wide, linearized	55.0	56.7
VGG-11 wide, linearized	61.0	61.7
Prior features (Oyallon et al.)	-	82.3
Random features (Recht et al.)	-	84.2
VGG-11 wide, standard	99.9	89.7
ResNet wide, standard	99.4	91.0

Linear vs. lazy vs. deep models

Theoretical arguments. Neural networks can be superior to kernel/fixed features methods, thanks to their adaptivity (Bach 2017).

Main references

- Jacot et al., *Neural Tangent Kernel: Convergence and Generalization in Neural Networks*. 2018.
- Du et al., *Gradient Descent Provably Optimizes Over-parameterized Neural Networks*. 2018.
- Bach. *Breaking the Curse of Dimensionality with Convex Neural Networks)*. 2017.