A PARTIAL THEORY OF WIDE NEURAL NETWORKS USING WC FUNCTIONS AND ITS PRACTICAL IMPLICATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a framework based on the theory of Polyak-Łojasiewicz functions to explain the properties of convergence and generalization of overparameterized feed-forward neural networks. We introduce the class of Well-Conditioned (WC) reparameterizations, which are closed under composition and preserve the class of Polyak-Łojasiewicz functions, thus enabling compositionality of the framework results which can be studied separately for each layer and in an architecture-neutral way. We show that overparameterized neural layers are WC and can therefore be composed to build easily optimizable functions. We expose a pointwise stability bound implying that overparameterization in WC models leads to a tighter convergence around a global minimizer. Our framework allows to derive quantitative estimates for the terms that govern the optimization process of neural networks. We leverage this aspect to empirically evaluate the predictions set forth by some relevant published theories concerning conditioning, training speed, and generalization of the neural networks training process. Our contribution aims to encourage the development of mixed theoretical-practical approaches, where the properties postulated by the theory can also find empirical confirmation.

1 INTRODUCTION

A staggering aspect about neural networks is that they are seemingly able to overfit the training sample and yet generalize to unseen data. Such a behaviour has been actually observed in other learning paradigms in overparameterized settings, such as with linear regression (Bartlett et al., 2020) and kernel methods (Belkin et al., 2018; Tsigler & Bartlett, 2020), but neural networks are certainly the model characterized by the most surprising predictive performance on real-world data.

Several recent works (Schoenholz et al., 2017; Jacot et al., 2018; Du et al., 2019; Yang et al., 2020b) have attempted to explain the theoretical underpinnings of why neural networks learn and generalize. Inspired by infinitely wide neural networks (Jacot et al., 2018; Lee et al., 2019), much of this research has focused on developing a theory for very wide networks, i.e. where the number of parameters $m = \Theta(n^\alpha)$ is polynomially bigger than the number of examples n[1](Du et al., 2018; Allen-Zhu et al., 2019), and more recently restricting to overparameterized networks (Li et al., 2018; Oymak & Soltanolkotabi, 2020), i.e. where $m > n$. While these theoretical results are being obtained under increasingly milder (and hence more realistic) overparametrization assumptions, they are typically expressed only in terms of asymptotic rates. As such it is difficult to determine whether these results hold in practice out of the box of the theoretical models and thus if they can explain the behaviour of real-world neural networks.

Another line of research has focused on strongly experimental work (Lee et al., 2020). However their work is directed to obtain empirical suggestions for practitioners, and does not try to measure quantities that could further inform other theories.

[1]This is a condition that is not usually satisfied in actual networks. Consider, for instance, VGG19 which has 144M parameters distributed on 19 layers and was trained on 1.3M images: already a quadratic polynomial would require an order of 1 trillion parameters per layer for VGG, which is clearly unattainable for current practice. On the other hand VGG19 is mildly overparameterized, i.e. the number of parameters per layer is slightly greater than the number of used images.

The reasoning underpinning this work is that an ideal theory for neural networks must be applicable to state-of-the-art architectures, must be tested quantitatively for its adherence to reality, must be able to give both convergence[2] and generalization[3] guarantees, and should decouple the network models from their optimization. As a first step towards this aim, we investigate a new practical theory of convergence and generalization of overparameterized neural networks, that provides some quantitative estimates for the terms that govern the networks' optimization process. The theoretical model's predictions are later compared to the experimentally measured quantities to check its explanatory power. Advantages of the proposed theory concern its adaptability to different feed-forward architectures (due to the generality of Polyak-Łojasiewicz functions), a clear separation between model and optimization, and its testability, which allows it to be disproved experimentally.

An important novelty in our paper concerns the exposition of a "partial" theory and the novel role of the experimental results. In fact, as the reader will later observe in Section 5 and Section 6, we expose arguments in favour of the fact that wide layers are well-conditioned (a term that will be introduced later) during at least part of their training. Such arguments can become a proof if we are willing to impose the additional restriction of layer width being polynomially bigger than the number of examples. Instead of focusing on what can be proven, we choose to test if the reasoning that underpins those theories does hold in practice in mildly overparameterized networks. We think that shifting the focus to identifying discrepancies between existing theories and experiments on the realistic neural architectures can result in tighter feedback loops, where experiments can inform the development of new theoretical hypothesis that are more aligned with the reality of the neural models.

Our framework is based on the theory of Polyak-Łojasiewicz (PL) functions (Polyak, 1963) as they provide a simple and general condition to prove convergence of an objective function to a global minimum via first-order methods (Karimi et al., 2016; Csiba & Richtárik, 2017; Guille-Escuret et al., 2021). In addition, Charles & Papailiopoulos (2018) provide stability bounds for PL functions, allowing us to formulate bounds on the generalization of a neural network based on the amount of optimization performed leveraging the stability approach of Elisseeff et al. (2005).

1.1 Our contribution

Starting from these results, in Section 2 we extend those on PL functions (Csiba & Richtárik, 2017; Guille-Escuret et al., 2021) to deal with different amounts of sensitivity in each layer rather than being constrained to consider a single parameter for the whole network. In Section 4 we introduce the class of *Well-Conditioned* (WC) reparameterizations, which are closed under composition and preserve the class of Polyak-Łojasiewicz functions, and which we believe could be of independent interest. In this way we provide a simple framework that cleanly separates the model from the optimization process and from the generalization bounds. By providing theoretical clues (and later, via empirical measurements) we show that overparameterized neural layers are WC functions in Section 5.

We thus establish a first approach to compute quantitative estimates of important quantities in network optimization. We show how such estimates can characterize the training progess of real feed-forward neural models providing, in Section 6, an empirical analysis that compares the predictions of the theory concerning conditioning, convergence and generalization with the observed behaviour of trained models. The empirical analysis enables us to spot phenomena that cannot be explained by existing theories, for mildly overparameterized networks. We believe that this feature provides the fundamentals for directing theoretical research towards uncovering unexplained aspects of real-world neural architectures.

1.2 Differences with other works

The use of Polyak-Łojasiewicz functions to study convergence and behaviour of neural networks has been introduced only recently by few literature works. Very recently, Liu et al. (2020) have leveraged PL functions to obtain asymptotic rates of convergence. This approach, however, needs to consider the Hessian of the whole network, and focuses on explaining the behaviour of the Neural Tangent Kernel (NTK) Jacot et al. (2018), while our WC functions provide a theory amenable to be applied on a layer-by-layer basis without needing to consider the network as an indissoluble

[2] i.e. it should prove that the training on sample data will converge to a global solution.

[3] i.e. it should prove that the found solution will generalize well to fresh data from the same distribution.

entity. Moreover our results hold for different convex losses, and we provide quantitative convergence rates and we empirically validate them, including also an analysis of the generalization estimates of the trained models. As for what concerns the empirical analysis on the properties of wide neural networks, Lee et al. (2020) recently performed a large-scale experiment to study the performance of finite-width networks compared with their infinite limits, as predicted by the NTK theory. Our approach differs from Lee et al. (2020) in that we focus on fine-grained (i.e. at the level of single optimization steps) adherence of the empirical behaviour to the theory presented in this paper, while Lee et al. (2020) restricts to comparing the final outcomes of the optimization process for various finite- and infinite-width models, and thus serves mainly to guide empirical practitioners.

2 POLYAK-ŁOJASIEWICZ CONDITION

In this section we introduce the background on PL functions that we use throughout the paper. We provide a small generalization of the PL condition which will be later used in the analysis of the optimization of a multilayered neural network, and we confirm known convergence results (Karimi et al., 2016) in this new setting.

We remark that the presented theory concerning convergence speeds can be seamless extended to the two-sided PL setting of Yang et al. (2020a), thereby extending its applications to minimax optimization problems, such as those present in Generative Adversarial Networks Goodfellow et al. (2014) optimization. For readability purposed we expose here only the basic theory.

We consider multivariate functions $f : \prod_{i\in I} X_i \to \mathbb{R}$, where X_i are Banach spaces and I is finite. We will denote $X = \prod_i X_i$, $x \in X$ and will write $x_i \in X_i$ for the i-th component of x; given a function $f : X \to \mathbb{R}$ we will denote by $Df_x : X \to \mathbb{R}$ its Fréchet differential at x, by $D_i f_x : X_i \to \mathbb{R}$ its partial differential, by $\|Df_x\|$ the operator norm, and by $\nabla f(x) \in X^*$ is the dual vector corresponding to Df_x; the dependency of the quantities μ_i and L_i on points in the function domain Ω is omittied for notational simplicity; moreover we denote $\boldsymbol{\mu}$ the vector of $(\mu_i)_{i\in I}$.

Definition 1 (PL Condition). *Given a function $f : \prod_i X_i \to \mathbb{R}$ we say that f is $\boldsymbol{\mu}$-PL on Ω iff*

$$\forall x \in \Omega \quad \frac{1}{2}\sum_i \frac{1}{\mu_i}\|D_i f_x\|^2 \geq f(x) - f^* \tag{1}$$

where $f^* := \inf_{x\in\Omega} f(x)$.

The Polyak-Łojasiewicz condition basically states that the norm of the gradient at a point controls the minimality gap at the current point, and thus for this class of functions necessarily $\nabla f(x) = 0$ implies that x is a global optimum in Ω.

Definition 2 (Smoothness). *Given a function $f : \prod_i X_i \to \mathbb{R}$ we say that f is $\boldsymbol{L}$-smooth iff*

$$\forall x, y \in \prod_i X_i \quad f(x) - f(y) \leq \sum_i D_i f_y[x_i - y_i] + \frac{L_i}{2}\|x_i - y_i\|^2.$$

PL functions enjoy the useful property of exponential convergence to a point of minimum value via common first-order otimization methods (Karimi et al., 2016; Csiba & Richtárik, 2017; Guille-Escuret et al., 2021). In this work we only consider minimization via gradient descent, but the extension to other algorithms is standard and can be found in the cited papers. All the proofs of this Section are reported in Appendix A.

Lemma 1 (Convergence speed and radius for PL functions). *Let $f : \prod_i X_i \to \mathbb{R}$ be $\boldsymbol{\mu}$-PL and $\boldsymbol{L}$-smooth; the X_i be Hilbert spaces. Choose an initial point x_0 and let the sequence of iterates evolve according to the rule*

$$x_i^{k+1} = x_i^k - \frac{1}{L_i}\nabla_i f(x^k). \tag{2}$$

Letting $\gamma := 1 - \min_i \frac{\mu_i}{L_i}$, the optimality gap decreases exponentially following the formula

$$f(x^{k+1}) - f^* \leq \gamma\big(f(x^k) - f^*\big).$$

Moreover, the distance from the initial point is bounded by

$$\sum_i \frac{1}{L_i}\big\|x_i^{k+1} - x_i^0\big\| \leq \sqrt{2(f(x^0) - f^*)\left(\sum_i L_i^{-1}\right)}\frac{1}{1-\sqrt{\gamma}}$$

Additionally, PL functions theory encompasses convex optimization because of the following lemma.

Lemma 2 (Strongly convex functions are PL)**.** *Let $f : X \to \mathbb{R}$ be a τ-strongly convex function. Then f is τ-PL on every set $\Omega \subseteq X$, i.e. $\forall x \in \Omega \quad \|\nabla f(x)\|^2 \geq \tau(f(x) - f^*_\Omega)$ where $f^*_\Omega := \inf_{x\in\Omega} f(x)$.*

3 Stability results for Polyak-Łojasiewicz functions

In this section we recall the notion of stability and we introduce a result by Charles & Papailiopoulos (2018) that provides stability bounds with a PL empirical risk, which leads to a generalization bound.

We briefly recall standard notation from Elisseeff et al. (2005): given a labeled dataset $S = \{z_i = (x_i, y_i) \mid i = 1, \dots, n\}$ with examples sampled i.i.d. from a distribution $\mathcal{D}$ and a learning algorithm $\mathcal{A}$, let $w_S := \mathcal{A}(S)$ the algorithm's output on S[4]. Let ℓ be the loss function, and f the considered model. Then the empirical training error is defined by

$$R_S(w) = \frac{1}{|S|} \sum_{(x,y)\in S} \ell(f(w, x), x, y).$$

Generalization bounds can be obtained using the notion of pointwise hypothesis stability as follows.

Definition 3 (Pointwise stability, Elisseeff et al. (2005))**.** *An algorithm $\mathcal{A}$ has pointwise hypothesis stability β_n with respect to the loss function ℓ iff:*

$$\forall i \in \{1, \dots, m\} \quad \mathbb{E}_{S\sim\mathcal{D}^n, z\sim\mathcal{D}}[|\ell(f_S, z_i) - \ell(f_{S^{i,z}}, z_i)|] \leq \beta_n$$

where $f_S := f(w_S, \cdot)$ and $S^{i,z} = \{z_1, z_2, \dots, z_{i-1}, z, z_{i+1}, \dots, z_n\}$.

Lemma 3 (Generalization from pointwise stability, Elisseeff et al. (2005))**.** *Suppose $\mathcal{A}$ has pointwise stability β with respect to ℓ where $0 \leq \ell(w, z) \leq M$. Then for any δ, with probability at least $1 - \delta$ we have:*

$$|R_{\mathcal{D}}(w_S) - R_S(w_S)| \leq \sqrt{\frac{M^2 + 12Mn\beta}{2n\delta}}$$

To connect stability theory to PL models we make use of a result by Charles & Papailiopoulos (2018). In what follows we will denote by w^*_S a global minimizer towards which the algorithm is converging.

Lemma 4 (R_S PL implies pointwise stability, (Charles & Papailiopoulos, 2018, Theorem 3))**.** *Suppose that for every dataset S we have R_S is μ-PL, $|R_S(w_S) - R_S(w^*_S)| \leq \varepsilon_A$ and that $\ell \circ f$ is G-lipschitz with respect to w, then $\mathcal{A}$ has pointwise stability bounded by*

$$\beta_{ptw} \leq 2\sqrt{\varepsilon_{\mathcal{A}}}\sqrt{\frac{2G^2}{\mu}} + \frac{1}{n-1}\frac{2G^2}{\mu}$$

Notice that while the second term is always present, the first one depends on the amount of optimization performed by the algorithm, which vanishes for a perfect fitting of the training data, and which suggests that for PL models more overfitting implies more generalization.

The presence of the quantity $\frac{2G^2}{\mu}$ hints that the bigger the PL constant of the model, the better it is both for generalization and optimization. Moreover it is better if deviations of the model are small with respect to its parameters and inputs[5].

4 Well-Conditioned functions

In this Section we introduce the class of *Well-Conditioned* (WC) functions, that are closed under composition, and whose composition with a PL function generates another PL function. They provide the foundations for our theory since, under certain conditions, neural layers are WC (Section 5).

Let X, Y be Banach spaces, and let $T : X \to Y$ be a linear functional; denote its range by $R(T) \subseteq Y$, and define the preimage of a point y as $S(T, y) = \{x \in X \mid Tx = y\}$. The definition of a WC function is the following.

[4]Correponding to neural networks weights in our case.

[5]Lemma 4 asks for a small lipschitz coefficient, and Lemma 1 asks for a small smoothness coefficient.

Definition 4 (WC function). *Let a separately differentiable function $f : \prod_i X_i \to Y$ be given. We say that f is $\boldsymbol{\lambda}$-WC on a domain $\Omega \subseteq \prod_i X_i$ iff*

$$\forall \omega \in \Omega \quad \forall i \quad \forall y \in R(D_i f_\omega) \qquad \inf_{x_i \in S(D_i f_\omega, y)} \lambda_i \|x_i\| \leq \|y\|.$$

The condition can be understood as a bound on the minimum norm solution x of the inverse problem $Tx = y$ where $T := D_i f_x$ is the partial differential of the function at a given point. In this way we essentially bound from below the norm of the differential of f, and by chain rule we can bound from below the PL coefficient of the composition, obtaining the following lemmas (proofs in Appendix A).

Lemma 5 (Composition of a PL function with WC is PL). *Let $g : Y \to \mathbb{R}$ be μ-PL over $\Omega \subseteq Y$ and $f : \prod_i X_i \to Y$ be $\boldsymbol{\lambda}$-WC over $\prod_i M_i \subseteq \prod_i X_i$ and suppose they satisfy the inequality*

$$\sum_i \left\| Dg_{f(x)}|_{R(D_i f_x)} \right\|_{op}^2 \geq \alpha \|Dg(f(x))\|_{op}^2 \tag{3}$$

and $f(\prod_i M_i) = \Omega$. Then $h(\vec{x}) = g(f(\vec{x}))$ is $(\mu\alpha\boldsymbol{\lambda}^2)$-PL over $\prod_i M_i$ and $h^* = g^*_\Omega$.

Remark 1. *Condition (3) in Lemma 5 is satisfied in particular if there exists one variable x_i for which $D_i f_x$ is a dense subspace of Y for all x*[6].

Lemma 6 (Composition of a WC function with WC is WC). *Let $g : \prod_i Y_i \to Z$ be $(,_i \lambda_i)$-WC and $f_i : \prod_j X_{ij} \to Y_i$ be $(,_{ij} \mu_{ij})$-WC. Then their composition $h : \prod_{ij} X_{ij} \to Z$ defined by*

$$h(,_{ij} x_{ij}) = g(,_i f_i(,_j x_{ij}))$$

is $(,_{ij} \lambda_i \mu_{ij})$-WC. By $f(,_{i=1}^n x_i)$ we mean $f(x_1, \ldots, x_n)$.

Thus if we have a set of WC functions $f_1, \ldots, f_L$ satisfying condition (3), we can compose them together with a convex function g, to obtain a PL function $h = g \circ f_1 \circ \ldots \circ f_L$.

5 Wide neural network layers are Well-Conditioned

We now present an argument by Agarwal et al. (2020) that proves that conditioning at initialization improves exponentially with depth; we later show that it is possible to bound conditioning in a region of small distance from the initialization, and thus to prove that overparameterized layers of neural networks are WC functions[7]. We start the exposition giving a few definitions.

Definition 5 (Inverse Norm of a linear operator). *Given a linear operator $T : X \to Y$ define its inverse norm by*

$$\zeta(T) := \sup_{x \in X} \inf_{x' \in S(T, Tx)} \frac{\|x'\|}{\|Tx\|}$$

which coincides with the smallest λ for which T is $\frac{1}{\lambda}$-WC.

In order for the results to generalize over various architectures, we consider a layer as composed of a linear combination of a fixed non-linear part, toghether with some fixed M_{ij} that determines the available connections between inputs and outputs[8].

Definition 6 (Representation of a layer). *Let $A = \mathbb{R}^{m,k}$, $X = \mathbb{R}^k$ and $Y = \mathbb{R}^m$, $M_{ij} \in \mathbb{R}^{m,k}$ and $\phi : \mathbb{R} \to \mathbb{R}$ be a fixed function. Then define the function $f_M : A \times X^n \to Y^n$ by*

$$f_M(\alpha, x)_i^\nu = \sum_j \alpha_{ij} M_{ij} \phi(x_j^\nu).$$

We will now relate the WC coefficient of a neural network layer to its conditioning, which will allow us to later measure WC coefficients. Then we expose the arguments by Agarwal et al. (2020) that clarify why it is expected that conditioning (and WC coefficient) improves with depth.

In this context we only consider conditioning of a linear operator with respect to the euclidean norm; thus conditioning is the ratio between the highest and the lowest singular values of the operator.

[6] This is typically the case if a single layer has more neurons than the number of available datapoints.

[7] What we actually prove is that overparameterized layers of neural networks are very often (with respect to the randomness in their initialization) WC functions near their initialization.

[8] For FCN we will have $\forall i, j \quad M_{ij} = 1$; for CNN they will be set accordingly to the kernel size and stride.

WC coefficients. A neural layer has two WC coefficients: one associated to the layer parameters and one bound to its inputs, which allows us to chain WC coefficients using Lemma 6.

Parameter WC coefficient. The linear operator of interest is $T = D_\alpha f_M(\alpha, x)$ and by Definition 5 we want to compute $\zeta(T)$, since $\mathrm{WC}(T) = 1/\zeta(T)$. Definition 5 can be cast as the following problem: suppose that we are given a fixed $\vec{y} \in Y^n$ which is in the range of $f_M(\cdot, x)$; we want to solve

$$\min_\alpha \|\alpha\|_2 \text{ subject to } f_M(\alpha, \vec{x}) = \vec{y}\,. \tag{4}$$

To solve it, define the matrix $(F_x)_{ki,i'j} = \delta_{ii'} M_{i'j} \phi(x_j^k)$ so that $y_i^k = \sum_{i'j} (F_x)_{ki,i'j} \alpha_{i'j}$ and by the theory of minimum norm solutions the solution to Equation (4) is given by $\alpha = F_x^T (F_x F_x^T)^{-1} y$. Thence $\zeta(D_\alpha f_M(\alpha, x)) = \lambda_{\min}(F_x F_x^T)^{-1/2}$, which gives us a way to compute the WC coefficient and shows our interest in the eigenvalues of the Gram matrix $H_x := F_x F_x^T$.

Bounding the eigenvalues. We are able to say that conditioning of the Gram matrix H_x does get exponentially better with depth, using arguments by Agarwal et al. (2020); Daniely et al. (2016). Consider a network with activation function σ and assume inputs x_i have unitary norm and that their products $|\langle x_i, x_j \rangle| \leq 1 - \delta$ are bounded by one. Let us also define the dual activation[9]

$$\hat{\sigma}(\rho) = \mathbb{E}_{(X,Y)\sim\Sigma_\rho}[\sigma(X)\sigma(Y)] \quad \text{where } \Sigma_\rho = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}.$$

Assume that the initial weights are distributed accordingly to a gaussian distribution that preserves the forward variance, and consider each layer gram matrix $H_{ij}^L = \langle x_i^L, x_j^L \rangle$, where x_i^L is the i-th input to the L-th layer. Given the definition of $\hat{\sigma}$, it is clear that $H^{L+1} = \hat{\sigma}(H^L)$, where $\hat{\sigma}$ is applied to each entry of the previous layer matrix. Using the following lemma iteratively it is then possible to derive bounds on the smallest eigenvalues of the gram matrix H^L at initialization.

Lemma 7 (Eigenvalue lower bound lemma (Agarwal et al., 2020, Lemma 23))**.** *Let $H \in \mathbb{R}^{n\times n}$ a positive-definite matrix, i.e. $H \geq \delta I_n$ and all values $= 1$ on the diagonal. Let $f : \mathbb{R} \to \mathbb{R}$ be an analytic function whose series expansion in zero has only positive coefficients, and let $f[H]$ be the application of f to each entry of the matrix. Then $f[H] \geq (f(1) - f(1-\delta))I_n$.*

It is also possible to derive a bound on the non-diagonal entries of the gram matrix, provided that we can bound the amount of distortion that $\hat{\sigma}$ can induce: $M_\delta := \sup\{\hat{\sigma}(\rho) \mid |\rho| \leq 1 - \delta\}$. This estimate provides a way to bound the highest eigenvalue via Gershgorin Circle Theorems; and when combined with Lemma 7 it shows that conditioning improves with depth. For more details we refer the reader to Agarwal et al. (2020) and Daniely et al. (2016).

Input WC coefficient. Analogously to the parameters WC coefficients, we have $\zeta(D_x f_M(\alpha, x)) = \lambda_{\min}(R_x R_x^T)^{-1/2}$ where the matrix $(R_x)_{\ell i,kl} = \delta_{k\ell} \alpha_{ij} M_{ij} \phi'(x_j^k)$ is the linear relation between the derivatives and the change in Y^n.

It would be ideal to also prove that finite neural networks are very often well-conditioned at initialization. To do this one could use the fact that the matrix norm between the mean of H_x and the H_x of a single random sample depends weakly on a large number of variables (the α coefficients), so that McDiarmid (1989) inequality can be used to limit the two matrices' distance in operator norm with high probability the bound transfers to a bound on the smallest eigenvalue via the following lemma.

Lemma 8 (Inverse Norm Bound)**.** *Let $T, \overline{T} : X \to Y$ be two linear operators with the same range, i.e. such that $R(T) = R(\overline{T})$ and satisfying $\zeta(T)\left\|T - \overline{T}\right\|_{op} < 1$. Then we have*

$$\zeta(\overline{T}) \leq \frac{\zeta(T)}{1 - \zeta(T)\left\|T - \overline{T}\right\|_{op}}.$$

The proof of Lemma 8, a generalization of some results by Ma (2012), is reported in Appendix C.

[9]Notice that in the definition of the dual activation the mean over all possible network initializations is taken, which does correspond to the setting of "infinitely-wide" neural networks (Jacot et al., 2018) due to central limit theorems. In the experimental section we will analyze how these predictions change on finite networks.

As mentioned in the introduction, we provide only proof ideas for this part since similar reasonings can be found in many theoretical papers about very wide or infinite neural networks (Allen-Zhu et al., 2018; Du et al., 2019; Zou et al., 2020), and we do later experimentally measure WC coefficients of real networks at each epoch, thus ensuring that the tested networks are WC functions during the whole training. This also enables us to test whether the reasoning in proofs for very wide networks is able to explain the phenomena of good conditioning which is a shared similarity of very wide and mildly overparameterized networks.

We now state a lemma that allows us to prove well-conditioning near initialization if we know that the layer is well-conditioned at initialization, a result that holds for the whole training in the case of very wide neural networks due to the fact that wide networks' coefficients move very little from initialization. The lemma infact crucially relies on the traveled distance of a layer coefficients from their initialization, and is implicit in many of the cited works. Proof can be found in Appendix B.

Lemma 9 (Conditioning of a layer)**.** *Let $f_M(\alpha, x)$ be defined as before; suppose that there $\exists x_0 \in X^n$ such that $f_M(\cdot, x_0) : A \to Y^n$ has dense image in Y^n, and we have a bound on the inverse norm of both derivatives: $\zeta(D_\alpha f_M(x_0, \alpha)) \leq \frac{1}{\lambda_\alpha}$ and $\zeta(D_x f_M(x_0, \alpha)) \leq \frac{1}{\lambda_x}$.*

Then f_M is WC with coefficients $\lambda_\alpha - \|\alpha M\|_2 G_\phi \|x - x_0\|$ with respect to A and $\lambda_x - R_\phi \|\alpha M\|_\infty \|x - x_0\|$ with respect to X, where G_ϕ is the lipschitz constant of ϕ, and R_ϕ is a bound to the second derivative of ϕ, i.e. $R_\phi := \sup_{r \in \mathbb{R}} |\phi''(r)|$.

6 Empirical Analysis

In this section we discuss the settings and questions underlying our experiments; we then comment on the results for the different areas of inquiry (conditioning, convergence, generalization). Additional results and further observations are provided in Appendix F[10].

Experimental Setting We train several overparameterized FCNs on random subsets of the CIFAR10 dataset (Krizhevsky et al., 2009)[11]. The networks are initialized with Gaussian Kaiming initialization (He et al., 2015) to preserve the variance of activations in the forward pass; activation functions are normalized according to Agarwal et al. (2020). Input data is normalized such that $\|x_i\| = \sqrt{m}$[12], where m is the width of the first layer.

We focus on the following questions: (1) Is the conditioning theory (Agarwal et al., 2020) predictive of the measured conditioning? (2) How much does the lowest eigenvalue degrade with distance from initialization? (3) Is the PL theory predictive of optimization progress? (4) Is the generalization theory of PL functions (Charles & Papailiopoulos, 2018) predictive of generalization performance?

Conditioning at initialization We consider FCN networks consisting of 30 layers of varying widths (1000, 2000, 5000), different activation functions (ReLU and Tanh), over multiple numbers of randomly extracted examples (100, 200, 500, 1000), either renormalizing[13] after application of each layer or not, and averaging on three random seeds. The experiment has been run on a Tesla V100 PCIe 16GB GPU totaling a carbon footprint of 13Kg CO2 (calculated using CarbonFootprint).

Results are reported in Figure 1: we find a good adherence to the findings of Agarwal et al. (2020), despite the finite width; however we find that renormalization is essential for conditioning to continue decreasing steadily across layers (left plot)[14]; moreover we find that larger widths allow to reach a smaller conditioning for the same number of examples and also allow for a later departure between normalizing and non-normalizing behaviour (right plot). Basing on the adherence between the two behaviours in the initial layers, we speculate that layer normalization strategies may be removed from those without significant losses in accuracy. We leave a verification of this claim for future studies.

[10] Code for replicating the experiments is made available in supplemental material.

[11] Usage of a single dataset to validate the theory has been dictated by the heavy computational requirements of the experiments. Such a choice should nonetheless not impact the validity of the presented results, because our theory does not contain free parameters to be fitted on the dataset, and experiments are repeated for multiple random extractions of a subset of the data, thus leaving less chance for an overfit to the dataset.

[12] This is done to satisfy a scaled requirement of the unitary norm used in Agarwal et al. (2020).

[13] i.e. rescaling each datapoint such that its norm is the square root of the number of neurons in the layer.

[14] We think the phenomenon can be explained by small-width effects in the sampling of gaussian weights.

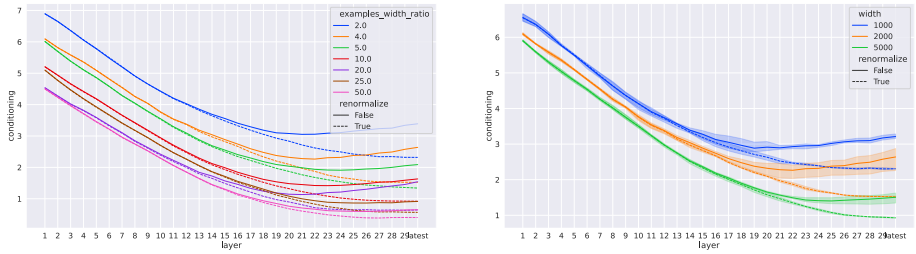

Figure 1: The first plot reports the mean parameter conditioning for ReLU FCNs at initialization as examples flow through the various layers; dotted lines represent renormalized networks. The values are reported as the natural logarithm of the conditioning of the H_x matrix defined in Section 5; line colors represent different examples-width-ratios of the tested networks. The second plot reports mean conditioning for 500 examples at varying widths; shaded regions denote 95% confidence intervals.

Training speed We consider networks of 6 and 9 FCN layers, of width 500, with different activation functions (ReLU, Tanh), and train them using full-batch gradient descent without momentum with mean-squared-error loss over 80 epochs, with various learning rates (0.0005, 0.001, 0.005, 0.01) and different number of randomly sampled examples (50, 100, 250, 500) over three random seeds. The experiment has been run on an A100 SXM4 40GB GPU totaling a carbon footprint of 12Kg CO2.

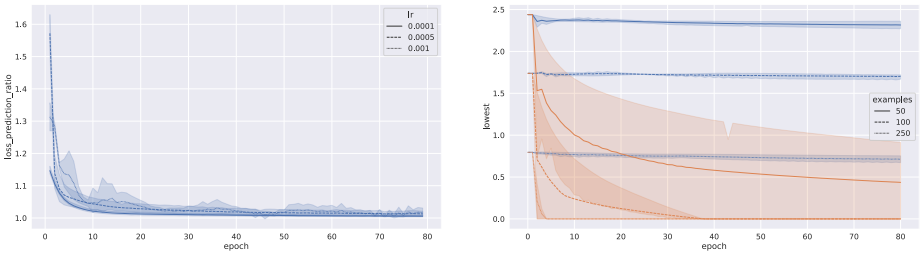

Figure 2: The first plot reports the ratio between the predicted and the measured loss decrease[15] at single epochs for 6-layers ReLU networks among different learning rates. The second plot reports in blue the progression of the lowest eigenvalue in the gram matrix H_x in the last layer, and in orange the lower bounds obtained by plugging the measured distance from initialization into Lemma 9.

Results are reported in Figure 2: we find that our upper bound on loss decrease given by Lemma 1[16] matches well the actual decreases in later epochs, while in initial epochs our estimate is too conservative (left plot). Concerning the measured lowest eigenvalues[17] of the gram matrix H_x, we find that they remain more or less constant during training (right plot) or tend to degrade gracefully (Figure 7), while the lower bound obtained via Lemma 9 degrades significantly, and even becomes vacuous in certain cases[18]. We emphasize the importance for theoretical research to look at possible explanations for this behaviour, which could greatly simplify a study of the properties of neural networks.

For what concerns the lowest eigenvalues of input conditioning (right plot of Figure 3), we find that they are almost zero for all the intermediate layers. This unfortunately does not enable us to use the more precise bounds on the decrease of the training loss that consider all layers' contributions.

[15] The "loss prediction ratio" refers to the ratio of the actual training loss and the bound calculated according to Lemma 1 where the PL and smoothness coefficients are empirically calculated on the network at the previous epoch. Local PL coefficients calculations are detailed in Appendix D.1.

[16] A detailed explanation of the measuring process for the PL coefficients is given in Appendix D.1.

[17] The lowest eigenvalues are exactly computed using power method on an implicit inverse of the H_x matrix.

[18] We remind the reader that other theoretical papers have used similar bounds (that depend on the distance of weights from their initialization) to explain the workings of very wide networks.

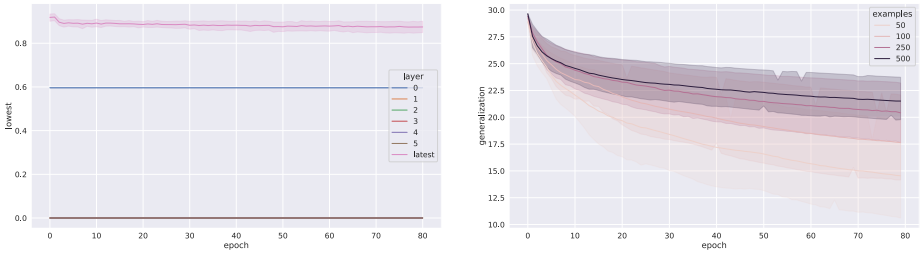

Figure 3: The first plot shows the lowest eigenvalue of the input conditioning matrix for the various layers. The right plot shows the generalization bounds obtained using the estimates in Lemma 4.

Generalization By measuring the local PL coefficients and estimates of the network lipschitz coefficients, we are able to use Lemma 4 and compute a generalization bound (depicted in the right plot of Figure 3). The plot shows the expected marked decrease with increasing optimization; despite this, the estimates are vacuous[19]. Further investigation on this issue is left to future work.

7 Conclusions

We have established and analyzed a theory of overparameterized neural networks based on the theory of PL functions, and suitably extended it. The experiments have partially proven the capacity of the theory on conditioning and on convergence speed to give meaningful results in real-world networks and data; they also have highlighted the need to consider other approaches for generalization theory and for explaining good conditioning exhibited by the networks.

This work is, to the extent of our knowledge, the first that tries to explicitly quantify abstract theories about the inner working of neural networks, and to compare the bounds obtained with real-world experiments to give a feedback to the developed theories. We hope that such an approach to the analysis of abstract theories might prove useful to direct the efforts of the theoretical community to advance understanding of neural modules.

7.1 Limitations and future work

Conditioning The conditioning theory by Agarwal et al. (2020) does generally conform to the experimental results of renormalized networks; we think that it would be greatly beneficial to expand the theory to also cover finite-width networks; moreover, the observed behaviour of the lowest eigenvalues during training currently remains unexplained.

PL Convergence The theory of optimization of PL functions seems to be able to explain the effective decrease in training loss apart for the initial epochs, in which the theory presents conservative bounds. Future work should focus on resolving the discrepancy between the fact that the measured input conditioning are extremely low and the known fact that training all layers gives better result than training only the last one.

PL Generalization The theory of generalization of PL functions via stability (Charles & Papailiopoulos, 2018) gives vacuous empirical bounds in the tested cases. We believe that a more adherent theory concerning generalization of neural networks should reconsider the role of intermediate layers as small deviations of random matrices, and not as arbitrary Lipschitz transformations, which can greatly impair the obtained bounds.

Proxy-PL We think that the notion of proxy-PLness by Frei & Gu (2021) would be extremely interesting to consider for generalization of the current theory (this work appeared when present work was under submission).

[19]The loss has a value between zero and one, while the predicted bounds are well over twenty.

REPRODUCIBILITY STATEMENT

Code to reproduce all the experiments presented in this paper, and the associated plots, is available in the supplemental materials (and will be publicly released post acceptance). Extra care has been taken to ensure reproducibility: a pinned conda environment file precisely describes the versions of the software that has been used to run the scripts, and DVC has been used to ensure the exact details of the commands to run the experiments and a cryptographic hash of their output has been saved. Functions in code are properly documented, and all computations accept an initial random seed to ensure reproducibility of the single runs.

REFERENCES

Naman Agarwal, Pranjal Awasthi, and Satyen Kale. A deep conditioning treatment of neural networks. *arXiv preprint arXiv:2002.01523*, 2020.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. On the convergence rate of training recurrent neural networks. *arXiv preprint arXiv:1810.12065*, 2018.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pp. 242–252. PMLR, 2019.

Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

Mikhail Belkin, Siyuan Ma, and Soumik Mandal. To understand deep learning we need to understand kernel learning. In *International Conference on Machine Learning*, pp. 541–549. PMLR, 2018.

Zachary Charles and Dimitris Papailiopoulos. Stability and generalization of learning algorithms that converge to global optima. In *International Conference on Machine Learning*, pp. 745–754. PMLR, 2018.

Dominik Csiba and Peter Richtárik. Global convergence of arbitrary-block gradient methods for generalized polyak-łojasiewicz functions. *arXiv preprint arXiv:1709.03014*, 2017.

Amit Daniely, Roy Frostig, and Yoram Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. *arXiv preprint arXiv:1602.05897*, 2016.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pp. 1675–1685. PMLR, 2019.

Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*, 2018.

Andre Elisseeff, Theodoros Evgeniou, Massimiliano Pontil, and Leslie Pack Kaelbing. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6(1), 2005.

Spencer Frei and Quanquan Gu. Proxy convexity: A unified framework for the analysis of neural networks trained by gradient descent. *arXiv preprint arXiv:2106.13792*, 2021.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27:2672–2680, 2014.

Charles Guille-Escuret, Manuela Girotti, Baptiste Goujaud, and Ioannis Mitliagkas. A study of condition numbers for first-order optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 1261–1269. PMLR, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pp. 8571–8580, 2018.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 795–811. Springer, 2016.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Jaehoon Lee, Lechao Xiao, Samuel S Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. *arXiv preprint arXiv:1902.06720*, 2019.

Jaehoon Lee, Samuel S Schoenholz, Jeffrey Pennington, Ben Adlam, Lechao Xiao, Roman Novak, and Jascha Sohl-Dickstein. Finite versus infinite neural networks: an empirical study. *arXiv preprint arXiv:2007.15801*, 2020.

Dawei Li, Tian Ding, and Ruoyu Sun. On the benefit of width for neural networks: Disappearance of bad basins. *arXiv*, pp. arXiv–1812, 2018.

Chaoyue Liu, Libin Zhu, and Mikhail Belkin. Toward a theory of optimization for over-parameterized systems of non-linear equations: the lessons of deep learning. *arXiv preprint arXiv:2003.00307*, 2020.

Hai Feng Ma, Shuang Sun, YuWen Wang, and Wen Jing Zheng. Perturbations of moore-penrose metric generalized inverses of linear operators in banach spaces. *Acta Mathematica Sinica, English Series*, 30(7):1109–1124, 2014.

Haifeng Ma. Construction of some generalized inverses of operators between banach spaces and their selections, perturbations and applications. 2012.

Colin McDiarmid. On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188, 1989.

Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

Boris Teodorovich Polyak. Gradient methods for minimizing functionals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 3(4):643–653, 1963.

Mark Rudelson and Roman Vershynin. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 62(12):1707–1739, 2009.

Samuel S Schoenholz, Justin Gilmer, Surya Ganguli, and Jascha Sohl-Dickstein. Deep information propagation. 2017.

Alexander Tsigler and Peter L Bartlett. Benign overfitting in ridge regression. *arXiv preprint arXiv:2009.14286*, 2020.

Junchi Yang, Negar Kiyavash, and Niao He. Global convergence and variance-reduced optimization for a class of nonconvex-nonconcave minimax problems. *arXiv preprint arXiv:2002.09621*, 2020a.

Zitong Yang, Yaodong Yu, Chong You, Jacob Steinhardt, and Yi Ma. Rethinking bias-variance trade-off for generalization of neural networks. In *International Conference on Machine Learning*, pp. 10767–10777. PMLR, 2020b.

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Gradient descent optimizes over-parameterized deep relu networks. *Machine Learning*, 109(3):467–492, 2020.

A PROOFS OF GENERALIZED PL AND WC LEMMAS

Lemma 1 (Convergence speed and radius for PL functions)**.** *Let $f : \prod_i X_i \to \mathbb{R}$ be $\boldsymbol{\mu}$-PL and $\boldsymbol{L}$-smooth; the X_i be Hilbert spaces. Choose an initial point x_0 and let the sequence of iterates evolve according to the rule*

$$x_i^{k+1} = x_i^k - \frac{1}{L_i}\nabla_i f(x^k). \tag{2}$$

Letting $\gamma := 1 - \min_i \frac{\mu_i}{L_i}$, the optimality gap decreases exponentially following the formula

$$f(x^{k+1}) - f^* \leq \gamma\big(f(x^k) - f^*\big).$$

Moreover, the distance from the initial point is bounded by

$$\sum_i \frac{1}{L_i}\big\|x_i^{k+1} - x_i^0\big\| \leq \sqrt{2(f(x^0) - f^*)\left(\sum_i L_i^{-1}\right)}\frac{1}{1-\sqrt{\gamma}}$$

Proof. We begin by proving the first formula:

$$\begin{aligned}
f(x_{k+1}) - f(x_k) &\leq -\sum_i \frac{1}{2L_i}\|\nabla_i f(x_k)\|^2 && \text{By smoothness and Equation (2)}\\
&= -\frac{1}{2}\sum_i \frac{\mu_i}{L_i}\frac{1}{\mu_i}\|\nabla_i f(x_k)\|^2\\
&\leq \left(\min_i \frac{\mu_i}{L_i}\right)\left(-\frac{1}{2}\sum_i \frac{1}{\mu_i}\|\nabla_i f(x_k)\|^2\right)\\
&\leq -\left(\min_i \frac{\mu_i}{L_i}\right)(f(x_k) - f^*) && \text{By definition of PL}
\end{aligned}$$

And thus we obtain

$$\begin{aligned}
f(x_{k+1}) - f^* &\leq (f(x_{k+1}) - f(x_k)) + (f(x_k) - f^*)\\
&\leq \left(1 - \min_i \frac{\mu_i}{L_i}\right)(f(x_k) - f^*).
\end{aligned}$$

Concerning the second equation we have:

$$\begin{aligned}
\sum_i \frac{1}{L_i}\left\|x_{k+1}^{(i)} - x_0^{(i)}\right\| &\leq \sum_i \sum_{t=1}^k \frac{1}{L_i}\|\nabla_i f(x_t)\|\\
&= \sum_{t=1}^k \sum_i \sqrt{\frac{2}{L_i}}\sqrt{\frac{\|\nabla_i f(x_t)\|^2}{2L_i}}\\
&\leq \sum_{t=1}^k \sqrt{2\sum_i L_i^{-1}}\sqrt{f(x_t) - f^*}\\
&\leq \sqrt{2\sum_i L_i^{-1}}\sum_{t=1}^k \gamma^{t/2}\sqrt{f(x_0) - f^*}\\
&\leq \sqrt{2(f(x_0) - f^*)\left(\sum_i L_i^{-1}\right)}\frac{1}{1-\sqrt{\gamma}}
\end{aligned}$$

and thus we have proved the thesis. □

Lemma 2 (Strongly convex functions are PL)**.** *Let $f : X \to \mathbb{R}$ be a τ-strongly convex function. Then f is τ-PL on every set $\Omega \subseteq X$, i.e. $\forall x \in \Omega \quad \|\nabla f(x)\|^2 \geq \tau(f(x) - f_\Omega^*)$ where $f_\Omega^* := \inf_{x\in\Omega} f(x)$.*

Proof. We recall that f being τ-strongly convex we may write

$$f(x) - f(y) \geq \langle \nabla f(y), x - y \rangle + \frac{\tau}{2}\|x - y\|^2$$

Now we take $\inf_{x\in\Omega}$ from both sides and we get:

$$\begin{aligned} f^* - f(y) &\geq \inf_{x\in\Omega} \langle \nabla f(y), x - y \rangle + \frac{\tau}{2}\|x - y\|^2 \\ &\geq \inf_{x\in X} \langle \nabla f(y), x - y \rangle + \frac{\tau}{2}\|x - y\|^2 \\ &= -\frac{1}{2\tau}\|\nabla f(y)\|^2 \end{aligned}$$

and we obtain the thesis. □

Lemma 5 (Composition of a PL function with WC is PL)**.** *Let $g : Y \to \mathbb{R}$ be μ-PL over $\Omega \subseteq Y$ and $f : \prod_i X_i \to Y$ be $\boldsymbol{\lambda}$-WC over $\prod_i M_i \subseteq \prod_i X_i$ and suppose they satisfy the inequality*

$$\sum_i \left\|Dg_{f(x)}|_{R(D_i f_x)}\right\|_{op}^2 \geq \alpha \|Dg(f(x))\|_{op}^2 \tag{3}$$

and $f(\prod_i M_i) = \Omega$. Then $h(\vec{x}) = g(f(\vec{x}))$ is $(\mu\alpha\boldsymbol{\lambda}^2)$-PL over $\prod_i M_i$ and $h^* = g^*_\Omega$.

Proof.

$$\begin{aligned} \frac{1}{2}\sum_i \frac{1}{\mu\alpha\lambda_i^2}\|Dh_{x_i}\|^2 &= \frac{1}{2}\sum_i \frac{1}{\mu\alpha\lambda_i^2}\left\|Dg_{f(x)}D_i f_x\right\|^2 \\ &= \frac{1}{2}\sum_i \frac{1}{\mu\alpha\lambda_i^2}\sup_{v_i\in X_i}\frac{\left\|Dg_{f(x)}[D_i f_x[v_i]]\right\|^2}{\|D_i f_x[v_i]\|^2}\frac{\|D_i f_x[v_i]\|^2}{\|v_i\|^2} \\ &\geq \frac{1}{2}\sum_i \frac{\lambda_i^2}{\mu\alpha\lambda_i^2}\sup_{y_i\in R(D_i f_x)}\frac{\left\|Dg_{f(x)}[y_i]\right\|^2}{\|y_i\|^2} \\ &\geq \frac{1}{2\mu}\frac{1}{\alpha}\sum_i \left\|Dg_{f(x)}|_{R(D_i f_x)}\right\|_{\text{op}}^2 \\ &\geq \frac{1}{2\mu}\left\|Dg_{f(x)}\right\|_{\text{op}}^2 \\ &\geq g(f(x)) - g^*_\Omega = h(x) - h^* \end{aligned}$$

because of the surjectivity of f on Ω. □

Lemma 6 (Composition of a WC function with WC is WC)**.** *Let $g : \prod_i Y_i \to Z$ be $(\_, _i\, \lambda_i)$-WC and $f_i : \prod_j X_{ij} \to Y_i$ be $(\_, _{ij}\, \mu_{ij})$-WC. Then their composition $h : \prod_{ij} X_{ij} \to Z$ defined by*

$$h(\_, _{ij}\, x_{ij}) = g(\_, _i\, f_i(\_, _j\, x_{ij}))$$

is $(\_, _{ij}\, \lambda_i\mu_{ij})$-WC. By $f(\_, _{i=1}^n\, x_i)$ we mean $f(x_1, \ldots, x_n)$.

Proof. It follows very easily from the definition: infact we can see that we want to bound the inverse solution norm:

$$\begin{aligned} \inf_{x_{ij}\in S(D_{ij}h_x, z)}\|x_i\| &= \inf_{y_i\in S(D_i g_{f(x)}, z)}\ \inf_{x_{ij}\in S(D_{ij}f_i(x), y_i)}\|x_{ij}\| \\ &\leq \inf_{y_i\in S(D_i g_{f(x)}, z)}\frac{1}{\mu_{ij}}\|y_i\| \\ &\leq \frac{1}{\mu_{ij}\lambda_i}\|z\| \end{aligned}$$

and we obtain the desired outcome. □

B PROOFS FOR WELL-CONDITIONING OF WIDE LAYERS

Lemma 9 (Conditioning of a layer)**.** *Let $f_M(\alpha, x)$ be defined as before; suppose that there $\exists x_0 \in X^n$ such that $f_M(\cdot, x_0) : A \to Y^n$ has dense image in Y^n, and we have a bound on the inverse norm of both derivatives: $\zeta(D_\alpha f_M(x_0, \alpha)) \leq \frac{1}{\lambda_\alpha}$ and $\zeta(D_x f_M(x_0, \alpha)) \leq \frac{1}{\lambda_x}$.*

Then f_M is WC with coefficients $\lambda_\alpha - \|\alpha M\|_2 G_\phi \|x - x_0\|$ with respect to A and $\lambda_x - R_\phi \|\alpha M\|_\infty \|x - x_0\|$ with respect to X, where G_ϕ is the lipschitz constant of ϕ, and R_ϕ is a bound to the second derivative of ϕ, i.e. $R_\phi := \sup_{r \in \mathbb{R}} |\phi''(r)|$.

Proof. We first bound the displacements of the partial derivatives:

$$\left\| f_M(\alpha, x)_i^k - f_M(\alpha, x')_i^k \right\|^2 \leq \left(\sum_j |\alpha_{ij} M_{ij}|^2 \right) \left(\sum_j \left\| \phi(x_j^k) - \phi(x_j'^k) \right\|^2 \right)$$

and thus we have $\sum_{k,i} \left\| f_M(\alpha, x)_i^k - f_M(\alpha, x')_i^k \right\|^2 \leq \|\alpha M\|_2^2 G_\phi^2 \|x - x'\|_2^2$.

We recall that for a symmetric matrix H we have $\|H\|_2 \leq \sqrt{\|H\|_1 \|H\|_\infty} = \|H\|_1$, and that

$$\frac{\partial f_\nu^r}{\partial x_i^k \partial x_j^\ell} = \alpha_{\nu j} M_{\nu j} \delta_{ij} \delta_{rk\ell} \phi''(x_i^k)$$

thus we can bound the displacement of the kernel derivatives by integrating:

$$\begin{aligned}
\left\| \left(D_x f_M(\alpha, x)_i^k - D_x f_M(\alpha, x')_i^k \right)[\delta x] \right\| &\leq \int_0^1 \mathrm{d}t \left\| D_{xx}^2 f_M(\alpha, x_t)_i^k [\delta x, x - x'] \right\| \\
&\leq \|x - x'\| \|\delta x\| \sup_t \left\| D_x^2 x f_M(\alpha, x_t)_i^k \right\|_{2 \to 2} \\
&\leq \|\alpha M\|_\infty \|x - x'\| \|\delta x\| \sup_{r \in \mathbb{R}} |\phi''(r)|
\end{aligned}$$

where $x_t = x_0 + t(x - x_0)$.

The two inequalities with Lemma 8 allow us to bound $\zeta(D f_M(\alpha, x))$ for x near x_0 and for any α:

$$\begin{aligned}
\zeta(D_\alpha f_M(\alpha, x)) &\leq \frac{1/\lambda_\alpha}{1 - \frac{1}{\lambda_\alpha} \|D_\alpha f_M(\alpha, x) - D_\alpha f_M(\alpha, x_0)\|} \leq \frac{1}{\lambda_\alpha - G_\phi \|\alpha M\|_2 \|x - x_0\|} \\
\zeta(D_x f_M(\alpha, x)) &\leq \frac{1/\lambda_x}{1 - \frac{1}{\lambda_x} \|D_x f_M(\alpha, x) - D_x f_M(\alpha, x_0)\|} \leq \frac{1}{\lambda_x - R_\phi \|\alpha M\|_\infty \|x - x_0\|}
\end{aligned}$$

because of hypothesis (1). □

C PROOF OF INVERSE NORM BOUND LEMMA

In this appendix we will give a proof of the Inverse Norm Bound Lemma. The one exposed here is a generalization of the one in Ma (2012); Ma et al. (2014).

Lemma 8 (Inverse Norm Bound)**.** *Let $T, \overline{T} : X \to Y$ be two linear operators with the same range, i.e. such that $R(T) = R(\overline{T})$ and satisfying $\zeta(T) \left\| T - \overline{T} \right\|_{op} < 1$. Then we have*

$$\zeta(\overline{T}) \leq \frac{\zeta(T)}{1 - \zeta(T) \left\| T - \overline{T} \right\|_{op}}.$$

We first recall the definition of $\zeta(T)$:

Definition 5 (Inverse Norm of a linear operator)**.** *Given a linear operator $T : X \to Y$ define its inverse norm by*

$$\zeta(T) := \sup_{x \in X} \inf_{x' \in S(T, Tx)} \frac{\|x'\|}{\|Tx\|}$$

which coincides with the smallest λ for which T is $\frac{1}{\lambda}$-WC.

Given T we define an approximate norm inverse function (possibly non-linear, non-continous) $T_\varepsilon^M : R(T) \to X$ if it respects $\forall y \in R(T)$ we have $T_\varepsilon^M y \in S(T,y)$ such that $\left\|T_\varepsilon^M y\right\| \leq \varepsilon + \inf_{x \in S(T,y)} \|x\|$.

For every $\varepsilon > 0$ there obviously exist many such functions and for them it holds $TT_\varepsilon^M = \mathrm{Id}$, $\lim_{\varepsilon \to 0} \left\|T_\varepsilon^M y\right\| = \inf_{x \in S(T,y)} \|x\|$. For them we define

$$\left\|T_\varepsilon^M\right\| := \sup_{y \in R(T)} \frac{\left\|T_\varepsilon^M y\right\|}{\|y\|}$$

and we can observe that

$$\left\|T^M\right\| := \lim_{\varepsilon \to 0} \left\|T_\varepsilon^M\right\| = \sup_{y \in R(T)} \inf_{x \in S(T,y)} \frac{\|x\|}{\|y\|} = \zeta(T).$$

We begin by setting the notation for the following lemmas. We will consider two linear operators $T, \overline{T} : X \to Y$ and we will fix $b \in R(T)$, $\overline{b} \in R(\overline{T})$. If not differently specified, $\delta T = \overline{T} - T$, $\delta b = \overline{b} - b$, $\overline{x} \in S(\overline{T}, \overline{b})$, $x \in S(T, b)$, $\overline{x}_\varepsilon = \overline{T}_\varepsilon^M \overline{b}$, $x_\varepsilon = T_\varepsilon^M b$.

Definition 7 ((Ma, 2012, Definition 2.1.4)). *Given a linear operator $T : X \to Y$, define*

$$\gamma(T) := \inf \{\|Tx\| \mid x \in X, \operatorname{dist}(x, N(T)) = 1\}$$

Lemma 10 (Bounds for distance from Kernel, (Ma, 2012, Lemma 4.1.1))**.**

$$\forall x \in X \quad \left\|T_\varepsilon^M T\right\|^{-1} \left\|T_\varepsilon^M T x\right\| \leq \operatorname{dist}(x, N(T)) \leq \left\|T_\varepsilon^M\right\| \|Tx\|$$

$$\forall z \in R(T) \quad \frac{1}{\left\|T^M\right\|} \leq \gamma(T) \leq \frac{\left\|T_\varepsilon^M T\right\| \left\|T T_\varepsilon^M z\right\|}{\left\|T_\varepsilon^M z\right\|}$$

Proof. Notice that $T_\varepsilon^M T x \subseteq \left\{x' \mid Tx' = Tx, \|x'\| \leq \inf_{\tilde{x} \in S(T, Tx)} \|\tilde{x}\| + \varepsilon\right\}$. Thus we have $(\mathrm{Id} - T_\varepsilon^M T)x \in N(T)$ and it follows that

$$\operatorname{dist}(x, N(T)) \leq \left\|x - (\mathrm{Id} - T_\varepsilon^M T)x\right\| = \left\|T_\varepsilon^M T x\right\| \leq \left\|T_\varepsilon^M\right\| \|Tx\|$$

On the other hand for $y \in N(T)$ we have

$$\left\|T_\varepsilon^M T x\right\| = \left\|T_\varepsilon^M T (x - y)\right\| \leq \left\|T_\varepsilon^M T\right\| \|x - y\|$$

and thus

$$\operatorname{dist}(x, N(T)) = \inf_{y \in N(T)} \|x - y\| \geq \left\|T_\varepsilon^M T\right\|^{-1} \left\|T_\varepsilon^M T x\right\|$$

Now we have $\gamma(T) = \inf_{x \in X} \frac{\|Tx\|}{\operatorname{dist}(x, N(T))}$ and thus we obtain from $\operatorname{dist}(x, N(T)) \leq \left\|T_\varepsilon^M\right\| \|Tx\|$:

$$\gamma(T) \geq \inf_{x \in X} \frac{\|Tx\|}{\left\|T_\varepsilon^M\right\| \|Tx\|} = \frac{1}{\left\|T_\varepsilon^M\right\|}$$

and from the other inequality for $\operatorname{dist}(x, N(T))$ we get

$$\gamma(T) \leq \left\|T_\varepsilon^M T\right\| \inf_{x \in X} \frac{\|Tx\|}{\left\|T_\varepsilon^M T x\right\|} \leq \frac{\left\|T_\varepsilon^M T\right\| \left\|T T_\varepsilon^M z\right\|}{\left\|T_\varepsilon^M z\right\|}$$

for $x := T_\varepsilon^M z$ and thus the inequality holds $\forall z \in R(T)$. □

Lemma 11 (Bounds for distance between inverse solution sets, (Ma, 2012, Lemma 4.1.4))**.** *Let $\overline{x} \in S(\overline{T}, \overline{b})$. Then we have*

$$\|T\|^{-1} \|\delta T \overline{x} - \delta b\| \leq \operatorname{dist}(\overline{x}, S(T, b)) \leq \left\|T^M\right\| \|\delta T \overline{x} - \delta b\|$$

Proof. Notice first that $S(T,b) = T_\varepsilon^M b + N(T)$, thus $\operatorname{dist}(\overline{x}, S(T,b)) = \operatorname{dist}(\overline{x} - T_\varepsilon^M b, N(T))$. Using Lemma 10 we have

$$\operatorname{dist}(\overline{x}, S(T,b)) \leq \left\|T_\varepsilon^M\right\| \left\|T(\overline{x} - T_\varepsilon^M b)\right\| = \left\|T_\varepsilon^M\right\| \|\delta T \overline{x} - \delta b\|$$

because $T = \overline{T} - \delta T$ and $T T_\varepsilon^M = \mathrm{Id}$. By taking the limit on $\varepsilon \to 0$ we obtain the equation.

On the other hand we can observe that for $y \in N(T)$ we have

$$\left\|T(\overline{x} - T_\varepsilon^M b)\right\| = \left\|T(\overline{x} - T_\varepsilon^M b - y)\right\| \leq \|T\| \left\|\overline{x} - T_\varepsilon^M b - y\right\|$$

and thus by taking the $\inf$ over $y \in N(T)$ we get

$$\operatorname{dist}(\overline{x}, S(T,b)) \geq \|T\|^{-1} \left\|T(\overline{x} - T_\varepsilon^M b)\right\| = \|T\|^{-1} \|\delta T \overline{x} - \delta b\|$$

as desired. □

Lemma 12. *If* $\left\|T^M\right\| \|\delta T\| < 1$ *and* $R(T) = R(\overline{T})$ *then*

$$\left\|\overline{T}^M\right\| \leq \frac{\left\|T^M\right\|}{1 - \|T^M\| \|\delta T\|}$$

Proof. If $\left\|T^M\right\| \|\delta T\| < 1$ then $\forall \varepsilon < \varepsilon_0$ it holds $\left\|T_\varepsilon^M\right\| \|\delta T\| < 1$.

Now given any $\overline{x} \in S(\overline{T}, \overline{b})$, we can find $x_\varepsilon \in S(T,b)$ such that

$$\|\overline{x} - x_\varepsilon\| \leq \operatorname{dist}(\overline{x}, S(T,b)) + \varepsilon \leq \left\|T^M\right\| \|\delta T \overline{x} - \delta b\| + \varepsilon$$

where the last inequality holds because of Lemma 11. Clearly such x_ε is such that $\|x_\varepsilon\| \geq \left\|T_\varepsilon^M b\right\| - \varepsilon$ by the definition of T_ε^M.

Consider now the following expression

$$\begin{aligned}
\frac{\left\|T_\varepsilon^M b\right\|}{\|b\|} - \left\|\overline{T}_\varepsilon^M\right\| &\leq \frac{\left\|T_\varepsilon^M b\right\| - \left\|\overline{T}_\varepsilon^M b\right\|}{\|b\|} && \text{By the definition of operator norm} \\
&\leq \frac{\|x_\varepsilon\| + \varepsilon - \|\overline{x}\|}{\|b\|} && \text{Where } \overline{x} := \overline{T}_\varepsilon^M b \text{ and } x_\varepsilon \text{ as above said} \\
&\leq \frac{\varepsilon + \|x_\varepsilon - \overline{x}\|}{\|b\|} \\
&\leq \frac{2\varepsilon + \left\|T^M\right\| \|\delta T\| \|\overline{x}\|}{\|b\|} \\
&\leq \frac{2\varepsilon}{\|b\|} + \left\|T^M\right\| \|\delta T\| \left\|\overline{T}^M\right\|
\end{aligned}$$

and by taking the sup on b such that $\|b\| = 1$, and then the limit for $\varepsilon \to 0$ we obtain

$$\left\|T^M\right\| - \left\|\overline{T}^M\right\| \leq \left\|T^M\right\| \|\delta T\| \left\|\overline{T}^M\right\|$$

which can easily be rearranged into the wanted equation. □

D Details of the Experiments

We reserve this section to give additional commands and details about the experimental setup.

D.1 Measurement of the PL coefficients

Recall that in Definition 1 multiple PL coefficients are present, one for each space X_i (which correspond in practice to the different layers of the netowrk). As detailed right above the definition, the quantities μ_i do depend on the point, and we measure them by computing the smallest eigenvalue of the linearization of the network function.

Such linearizations are computed separately for the weights of each layer: for the last layer the smallest eigenvalue is directly computed, while for earlier layers we rely on the characterization of their linearization as a composition of many linearization: that one of the last layer and R_x for each intermediate layer.

In this way we can apply the estimate $\sigma_*(AB) \geq \sigma_*(A)\sigma_*(B)$ where by σ_* we denote the smallest singular value of the matrix, and thus we obtain μ_n estimated via linearization of the last layer, $\mu_{n-1} \geq \mu_n * \zeta(R_x R_x^T)$ and so on.

By this means we unfortunately do only obtain marginal improvements over just using the last layer PL coefficient, but a direct measurement of the smallest eigenvalue of the linearizations for earlier layers was very difficult to do at the time using deep learning frameworks. We intend to measure such earlier layers' PL coefficients directly in future experiments by using some (still experimental) features of functorch.

Actual calculations In practice we calculate explicitly the H matrix, from which the Gram Matrices can be calculated as $G = HH^T$. We proceed by performing a QR decomposition of H such that $G = QRR^TQ^T$ and thus extremal values of G correspond to extremal eigenvalues of RR^T because of the orthogonality of Q and these can be easily calculated iteratively: the maximal eigenvalue is calculated using power iteration method on RR^T and the minimal eigenvalue is calculated by power iteration on the inverse matrix, obtained in implicit form (performing first a minimum norm solution problem with the matrix R and later a least square solution with R).

E Empirical Bounds Calculations

We provide many small calculations which are useful in deriving explicit numeric constants in the empirical neural network study.

In what follows we define the smoothness and lipschitz constant of a function $f : X \to Y$ by:

$$\|f(x) - f(x')\| \leq G_f \|x - x'\|$$
$$\|\nabla f(x) - \nabla f(x')\| \leq L_f \|x - x'\|$$

Lemma 13 (Smoothness Constant and Composition)**.** *Let $f : X \to Y$ and $g : Y \to Z$, and define $h(x) = g(f(x))$. Then we have $G_h \leq G_g G_f$ and $L_h \leq G_f^2 L_g + G_g L_f$.*

Proof. The result is well-known for the lipschitz constant, and for the smoothness constant we have:

$$\begin{aligned}\|\nabla h_x - \nabla h_{x'}\| &= \left\|\nabla g_{f(x)} \nabla f_x - \nabla g_{f(x')} \nabla f_x + \nabla g_{f(x')} \nabla f_x - \nabla g_{f(x')} \nabla f_{x'}\right\| \\ &\leq \|\nabla f_x\| \left\|\nabla g_{f(x)} - \nabla g_{f(x')}\right\| + \left\|\nabla g_{f(x')}\right\| \|\nabla f_x - \nabla f_{x'}\| \\ &\leq \left(G_f^2 L_g + G_g L_f\right) \|x - x'\|\end{aligned}$$

□

Lemma 14. *A Layer $y_i = \sum_j \alpha_{ij} M_{ij} \phi(x_j) = (\alpha \phi(x_\cdot))_i$ Lipschitz's constant with respect to x is $\leq \|\alpha M\|_{2\to 2} G_\phi$, and its Smoothness constant with respect to x is $\leq \|\alpha M\|_\infty \max |\phi''|$.*

On the other hand the smoothness constant with respect to α is zero, and the lipschitz constant is $\leq \lambda_{max}(F_x F_x^T)^{-1/2}$, where $(F_x)_{ki,i'j} = \delta_{ii'} M_{i'j} \phi(x_j^k)$.

Proof. We begin by calculating the lipschitz constant with respect to x:

$$\|y - y'\|^2 = \|\alpha(\phi(x_\cdot) - \phi(x'_\cdot))\| \leq \|\alpha\|_{2\to 2} G_\phi \|x_\cdot - x'_\cdot\|$$

and for the smoothness constant we observe that

$$\frac{\partial y_i}{\partial x_m \partial x_n} = \delta_{mn} \sum_j \alpha_{ij} M_{ij} \phi''(x_m) \delta x_m \delta x_n$$
$$\max \left\|\nabla_x^2 y\right\|_{2\to 2} \leq \max \left\|\nabla_x^2 y\right\|_{1\to 1} \leq \|\alpha M\|_\infty \max |\phi''|$$

The smoothness constant with respect to α is clearly zero since the function is linear in α, and the lipschitz constant can easily be found since the layer is a matrix multiplication. □

Lemma 15. *The mean-squared-error loss has $G \leq 2MSE(x)$, $L \leq N$, where N is the number of dimensions. Moreover it is a 1-strongly convex function.*

Proof. Recall that for the mean-squared-error loss we have:

$$\ell(x, y) = \frac{1}{2} \sum_i (x_i - y_i)^2$$
$$\frac{\partial \ell}{\partial x_n} = x_n - y_n$$
$$\frac{\partial^2 \ell}{\partial x_n \partial x_m} = \delta_{nm}$$

and thus we obtain that $\sum_n \left| \frac{\partial \ell}{\partial x_n} \right|^2 = 2\ell(x, y)$, $\sum_{n,m} \left| \frac{\partial^2 \ell}{\partial x_n \partial x_m} \right|^2 = N$.

It is a 1-strongly convex function because the Hessian is always positive definite and $H \geq I$. □

Lemma 16. *The cross-entropy loss has $L, G \leq 2$ and is a weakly convex function.*

Proof. Recall that cross-entropy satifyies

$$\ell(x, k) = -x_k + \log \left(\sum_j e^{x_j} \right)$$
$$\frac{\partial \ell}{\partial x_n}(x, k) = -\delta_{n,k} + \frac{e^{x_n}}{\sum_j e^{x_j}}$$
$$\frac{\partial^2 \ell}{\partial x_m \partial x_n}(x, k) = \frac{\delta_{mn} e^{x_n} \left(\sum_j e^{x_j} \right) - e^{x_m} e^{x_n}}{\left(\sum_j e^{x_j} \right)^2}$$

and one easily obtains that $\sum_n \left| \frac{\partial \ell}{\partial x_n}(x, k) \right|^2 \leq 2$, $\sum_{n,m} \left| \frac{\partial^2 \ell}{\partial x_m \partial x_n}(x, k) \right|^2 \leq 2$.

Weak convexity stems from the positivity of the hessian matrix: let us call $t_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$, then we have seen that $H_{nm} = \frac{\partial^2 \ell}{\partial x_m \partial x_n}(x, k) = \delta_{mn} t_n - t_n t_m$, and thus we have

$$u^T H u = \sum_{ij} H_{ij} u_i u_j = \sum_i t_i u_i^2 - \sum_{ij} t_i t_j u_i u_j = \sum_i t_i u_i^2 - \left(\sum_j t_j u_j \right)^2 \geq 0$$

by Cauchy-Schwartz inequality applied to $\sqrt{t_i}$ and $\sqrt{t_i} u_i$, since $\sum_i t_i = 1$. □

We have exposed the theory about strongly PL functions for simplicity, but we would like to highlight that to analyze the results of the experiments we have to rely also on the theory of weak PL functions which originated in the work of Csiba & Richtárik (2017). We expose the main results about weak PL functions, and refer the interested reader to the work.

Definition 8 (Weakly PL function, Csiba & Richtárik (2017))**.** *A function $f : X \to \mathbb{R}$ is weakly μ-PL on $\Omega \subseteq X$ if there $\exists x^* \in X^*$ the set of global minimizer such that*

$$\forall x, y \in \Omega \quad \|\nabla f(x)\| \|x - x^*\| \geq \sqrt{\mu}(f(x) - f^*)$$

where $f^* := \min_{x \in \Omega} f(x)$.

It can be easily observed that every convex function is weakly PL with $\mu = 1$. Moreover we have the following lemma on the decrease of the objective during the optimization process:

Lemma 17 (Gradient Descent on Weakly PL functions, (Csiba & Richtárik, 2017, Lemma 3))**.** *Given an L-smooth, weakly μ-PL function $f : X \to \mathbb{R}$ and chosen an initial point x_0, we let the sequence of iterates evolve according to the rule:*

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k).$$

Then the optimality gap decreases following the formula

$$f(x_{k+1}) - f^* \leq \left(1 - \frac{\mu(f(x_k) - f^*)}{2L\|x_k - x_*\|^2}\right)(f(x_k) - f^*).$$

Crucially we also have to show that the composition of a λ-WC function with a μ-weakly PL function does consist of a μ-weakly PL function. This does happen if we assume a small condition of non-degeneracy of the applied WC function.

Lemma 18. *Let $f : Y \to \mathbb{R}$ be a μ-weakly PL function with y^* the minimizer in its definition and $g : X \to Y$ be a λ-WC function such that $\exists q \in g^{-1}(y^*)$ satifying $\forall x \in X \quad \|x - q\| \geq \alpha\|g(x) - y^*\|$. Then the composition $h(x) = f(g(x))$ is $\mu\lambda^2\alpha^2$-weakly PL.*

Proof. The proof proceeds very similar in spirit to the one for PL functions:

$$\begin{aligned}\|\nabla h(x)\|\|x - q\| &= \|\nabla f(g(x)) \cdot \nabla g(x)\|\|x - q\| \\ &\geq \lambda\|\nabla f(g(x))\|\alpha\|g(x) - y^*\| \\ &\geq \sqrt{\mu\lambda^2\alpha^2}(h(x) - h^*)\end{aligned}$$

□

F Additional Experimental Results

We report here additional details about the experimental results. The focus of this additional analysis is on discussing phenomena which are not fully understood and on useful metrics that can guide a more principled developing of neural networks.

F.1 Scaling of the eigenvalues

We are interested in understanding the change in conditioning due to varying the width of the network and the number of examples used to train it. We compare the measured eigenvalues to the ones predicted by random matrix theory. In particular if we treat the matrix F_x as a random Gaussian matrix, using the result of Rudelson & Vershynin (2009) we would expect that

$$\begin{aligned}\lambda_{\text{max}} &\sim \sqrt{\text{width} \cdot \text{examples}} \\ \lambda_{\text{min}} &\sim \sqrt{\text{width}} - \sqrt{\text{examples}} \\ \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} &\sim \frac{1}{\sqrt{\frac{1}{\text{examples}}} - \sqrt{\frac{1}{\text{width}}}}\end{aligned}$$

and this is exactly what we can observe in Figure 4, which shows that such relation holds at every layer. This holds despite the fact that matrix F_x is not random, at least in the first layer.

F.2 Measures to detect chaotic behaviour

Given our findings, and particularly those in Figure 1, in principle one would like to avoid ending up training a network that is too deep for its width, thus creating the effect of raising the conditioning from some point onward. Measuring conditioning directly is inefficient, since it requires to solve the eigenvalue problem for very big matrices, where size depends both on the width of the network and on the number of examples, making this impractical for real datasets.

From the conditioning theory of Agarwal et al. (2020) we would expect off-diagonal entries of $F_x F_x^T$ to tend to zero as the signal propagates through the layers, hence we can measure this proxy

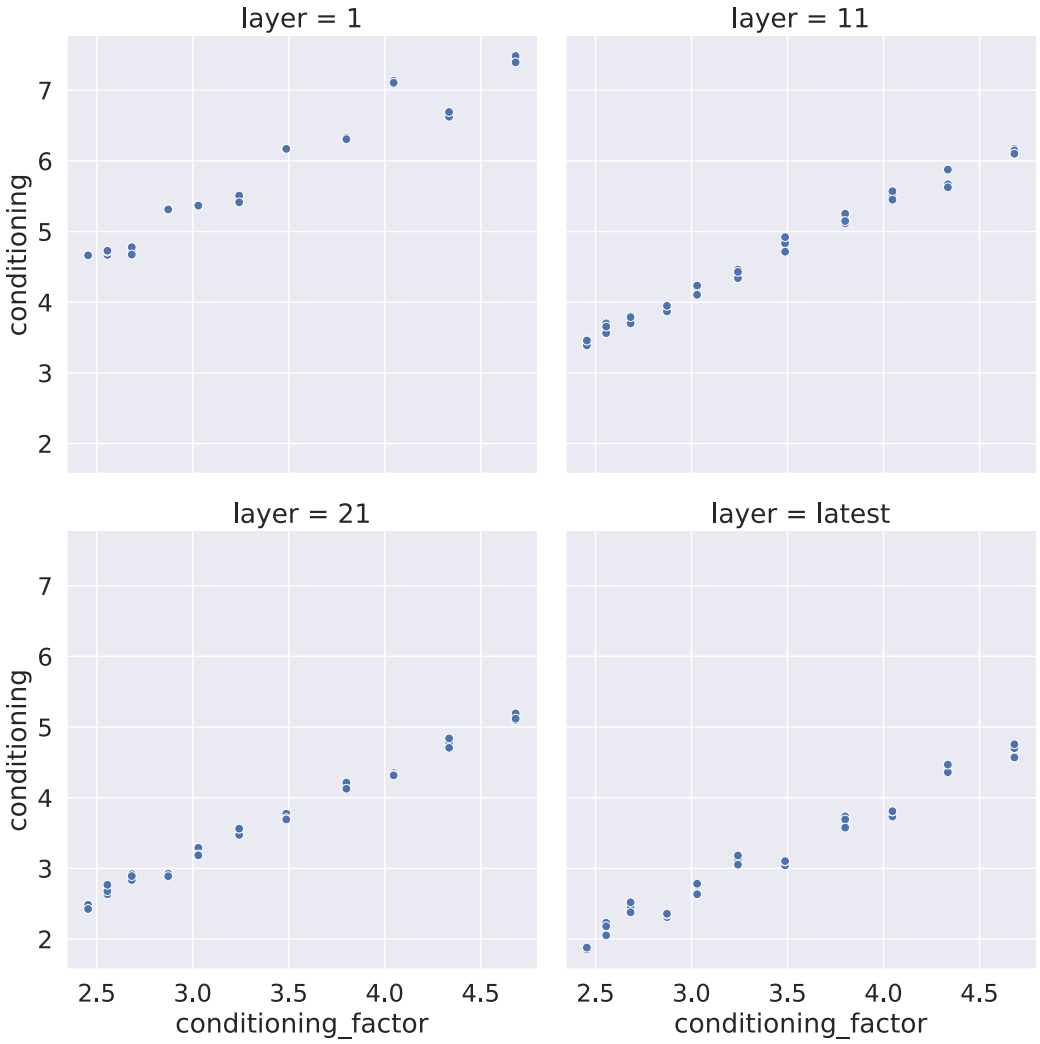

Figure 4: Conditioning plot with respect to the conditioning factor for Tanh activation networks.

information instead of measuring eigenvalues directly [20]. In Figure 5 the difference in behaviour between the normalized layers and the non-normalized ones is extremely evident, both in the values of the maximum value of the off-diagonal entries and of the maximum row sum of the matrix $F_x F_x^T$, which do align perfectly with the raise in conditioning that we have already observed in Figure 1.

F.3 Comparisons with infinite width networks

We compare the results obtained on conditioning with the ones calculated using the dual activation function theory by Agarwal et al. (2020); Daniely et al. (2016). Figure 6 shows the values for finite and for infinite width networks. For the finite width networks we report the measured highest and lowest eigenvalues, while for infinite width networks we report the upper bounds for the highest eigenvalue and the lower bound for the lowest eigenvalue. Thus the curves for infinite width networks are a little different than those for the other widths. Nonetheless the obtained plots show the validity of the theory, with its predictions closely following the observed values, and highlight the need for the study of an analogue finite-width theory to better compare theoretical results with mesurable ones.

[20] This has been used by Agarwal et al. (2020) themselves in their own experiment.

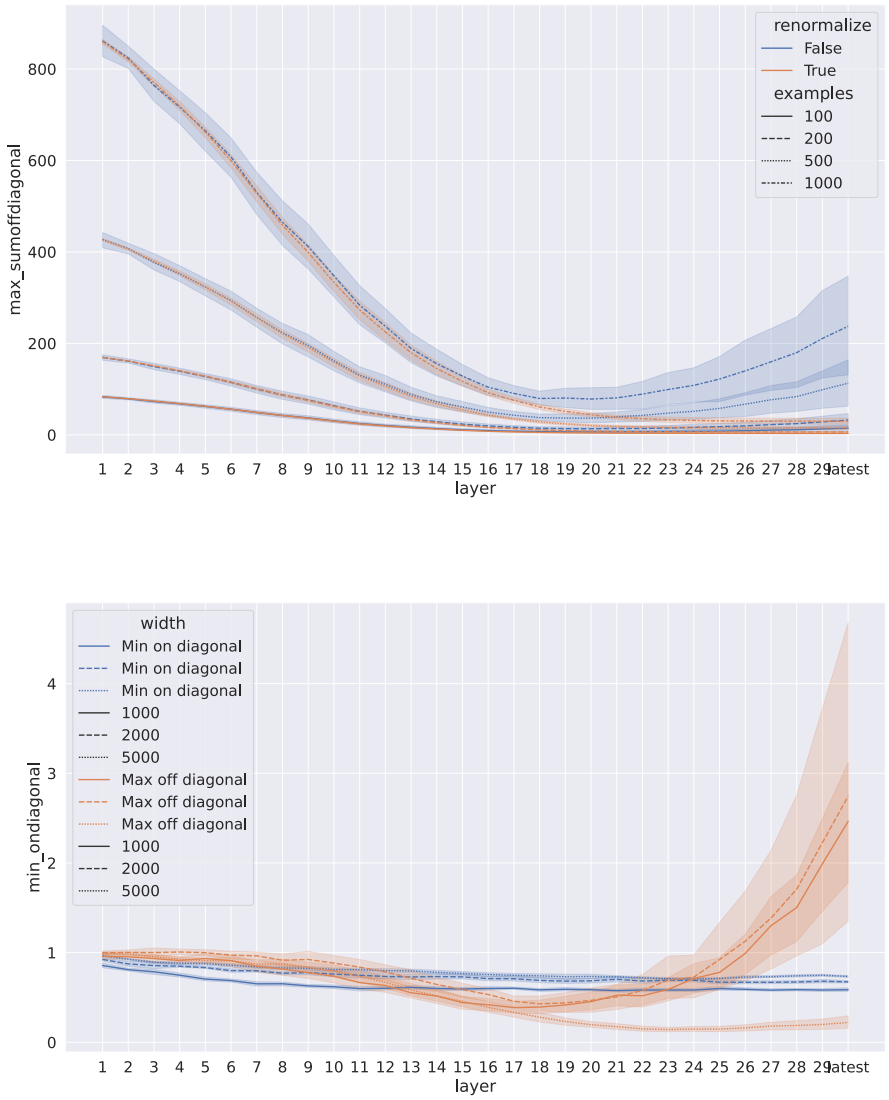

Figure 5: The first plot shows the maximum rowsum of the off diagonal entries of the matrix $F_x F_x^T$; the second plot shows the minimum value of the on-diagonal entry and the maximum value of the off-diagonal entries.

F.4 Eigenvalues during training in cross-entropy loss

We have seen in Figure 2 that the eigenvalues when considering a MSE loss remain mostly stable across training. What we instead observe for the cross-entropy loss (Figure 7) is that, especially at higher learning rates, we note an initial rapid increase in conditioning, followed by a steady decrease for both ReLU and Tanh activations.

This observations hints at the fact that there is some other aspect of the optimization of neural networks with respect to conditioning that is not fully understood, since we should reasonably expect that conditioning does continue to rise as the network weights are perturbed from their original positions.

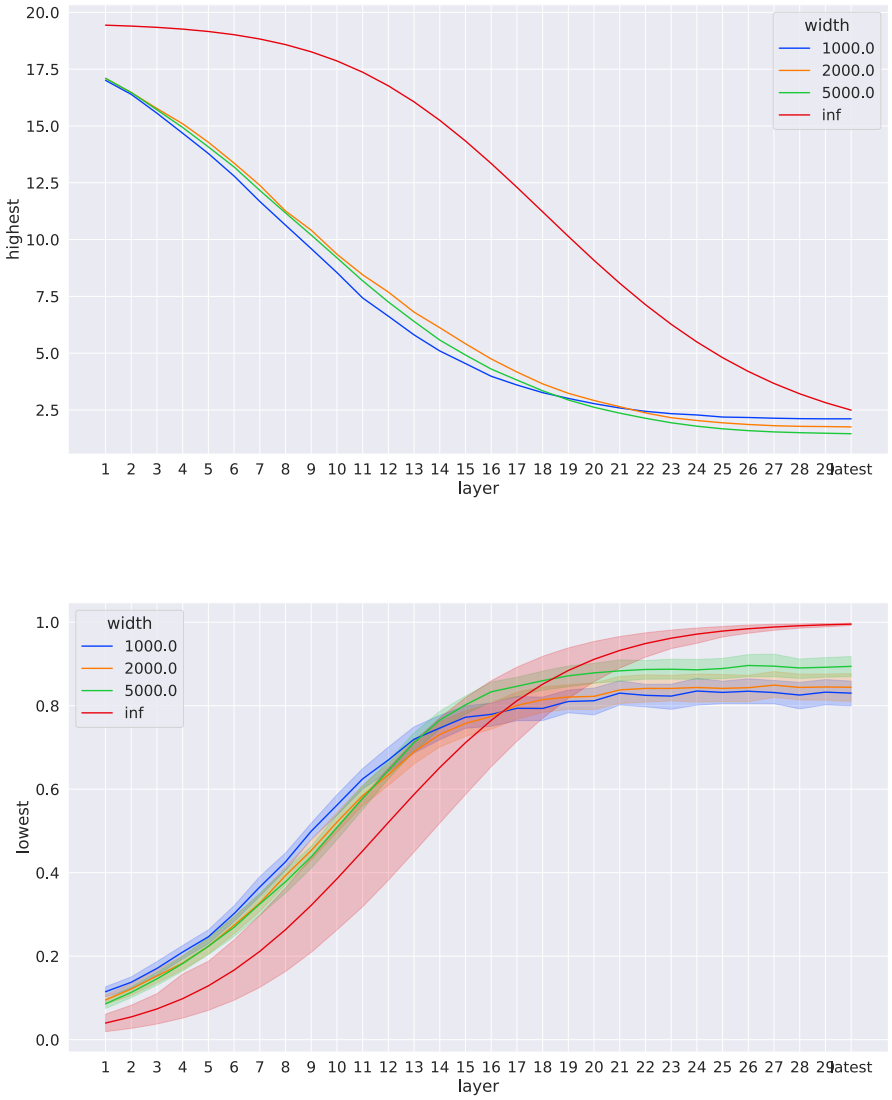

Figure 6: Plot of the normalized highest eigenvalue (top) and lowest eigenvalue (bottom) for the tested widths and for the theoretical prediction for a FCN with ReLU activation.

F.5 Prediction of training loss with cross-entropy

Using Lemma 17 for the cross-entropy loss and estimating $\|x_k - x^*\| \simeq 2\frac{f(x_k)-f^*}{G}$ where G is the Lipschitz constant of the function f, we obtain the results showed in Figure 8.

We can observe how the prediction accuracy are similar to what we have obtained for the mean-squared-error loss: our estimates are too conservative for the first epochs, but align well with successive epochs. Further analyses are needed to perfectly align theoretical predictions with experimental ones.

F.6 Cumulative prediction losses

We provide in Figure 9 and in Figure 10 plots for the cumulated loss prediction ratios, given as cumulative products of the single epoch ratios in time both for ReLU and Tanh networks trained under

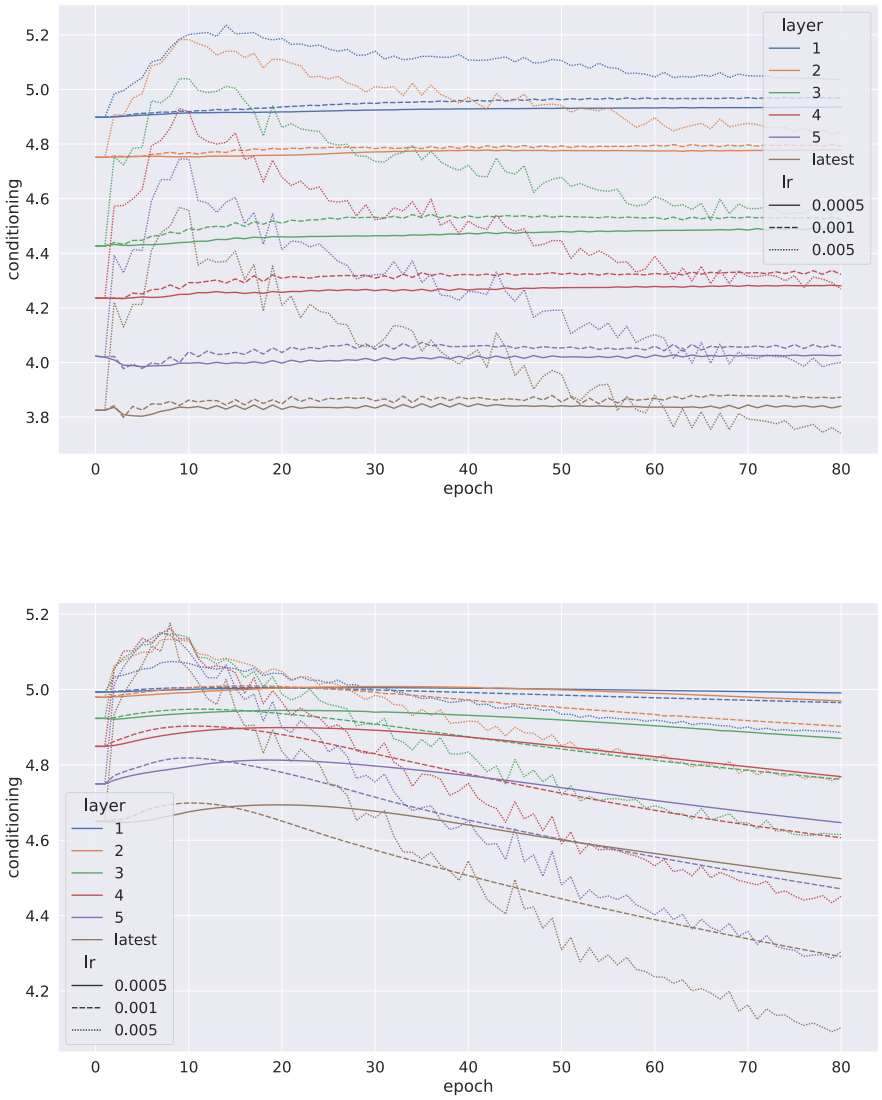

Figure 7: Plots of conditioning progression for ReLU (top) and Tanh (bottom) networks.

both Mean Squared Error objective and Cross-Entropy. The cumulative product gives a measure of the amount of uncertainty in the prediction of successive epochs.

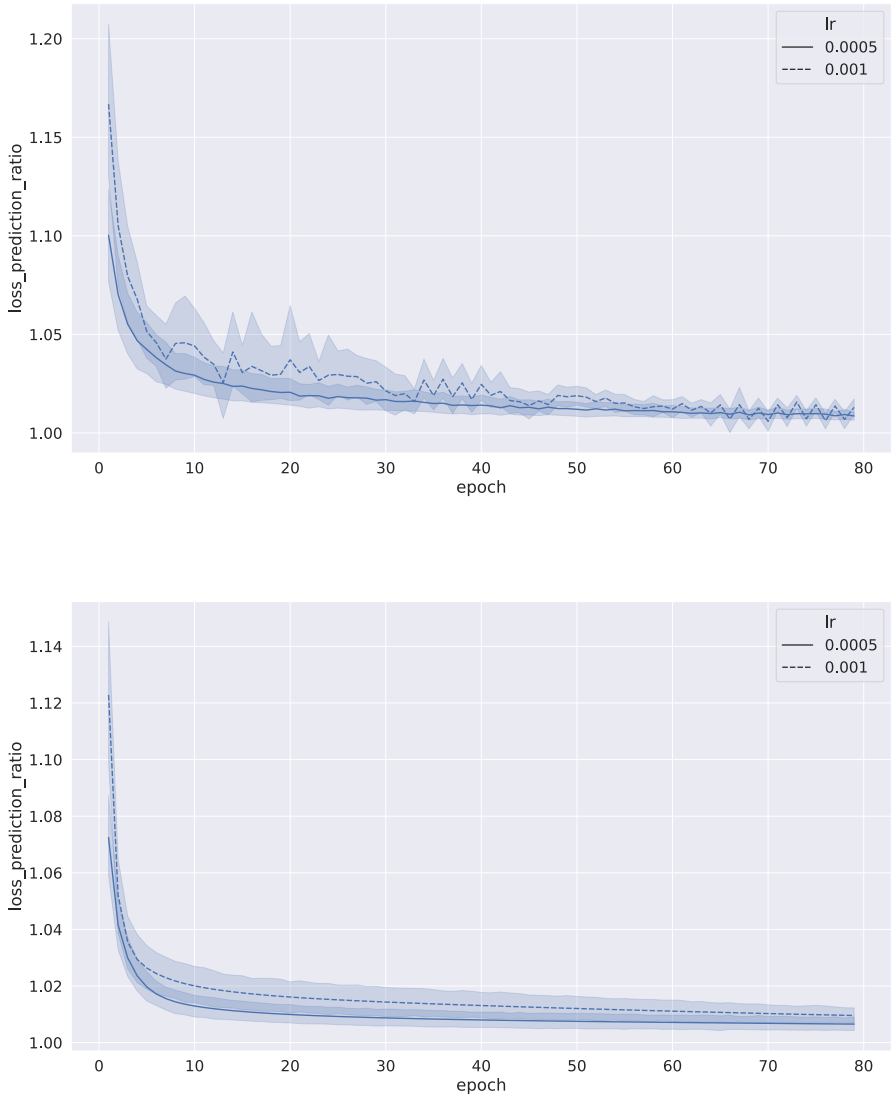

Figure 8: Plots of loss prediction ratio for ReLU and Tanh networks trained with the cross-entropy loss. The plots show the ratio between real loss and predicted loss at each epoch.

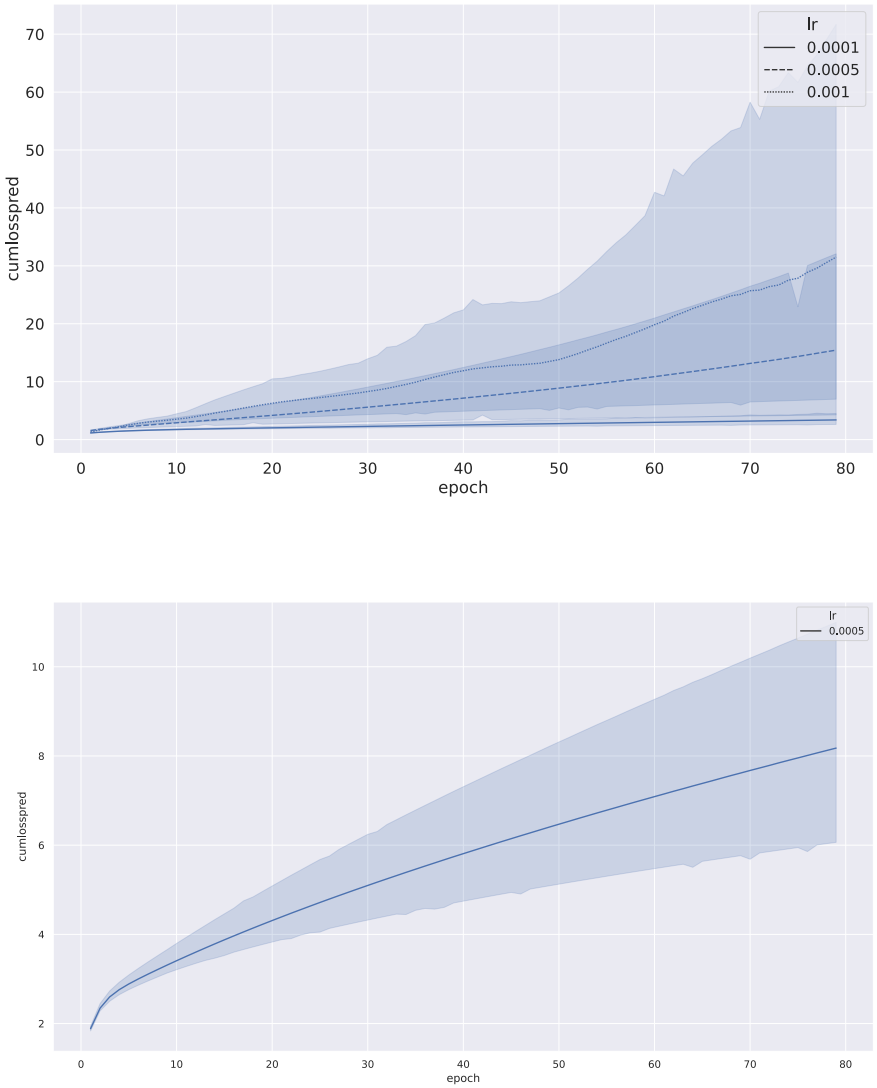

Figure 9: Cumulated loss prediction ratio for networks trained under Mean Squared Error, obtained as the cumulative product of the single epoch ratios. The first plot is for ReLU networks; the second plot details Tanh networks.

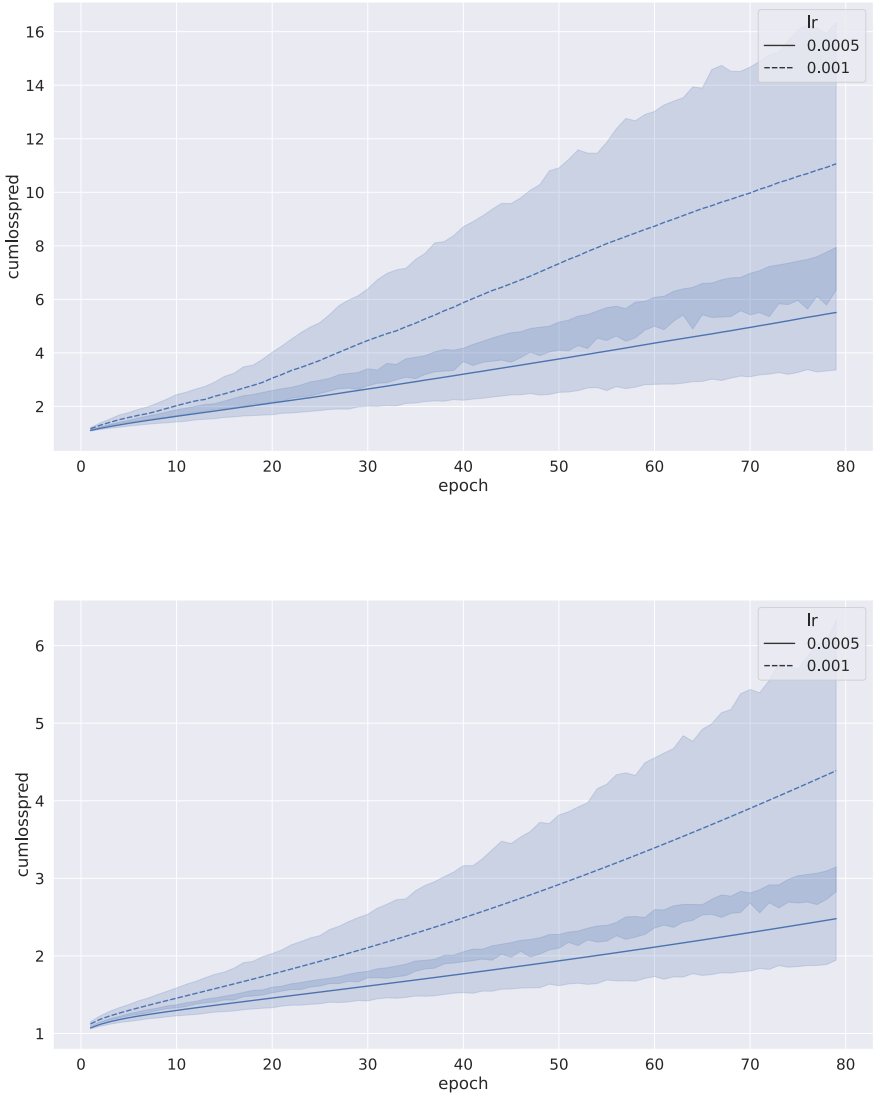

Figure 10: Cumulated loss prediction ratio for networks trained under Cross-Entropy, obtained as the cumulative product of the single epoch ratios. The first plot is for ReLU networks; the second plot details Tanh networks.