Stable ResNet

Soufiane Hayou*[1] **Eugenio Clerico*[1]** **Bobby He*[1]** **George Deligiannidis[1]**

Arnaud Doucet[1] **Judith Rousseau[1]**

Abstract

Deep ResNet architectures have achieved state of the art performance on many tasks. While they solve the problem of gradient vanishing, they might suffer from gradient exploding as the depth becomes large. Moreover, recent results have shown that ResNet might lose expressivity as the depth goes to infinity [Yang and Schoenholz, 2017, Hayou et al., 2019a]. To resolve these issues, we introduce a new class of ResNet architectures, called Stable ResNet, that have the property of stabilizing the gradient while ensuring expressivity in the infinite depth limit.

1 INTRODUCTION

The limit of infinite width has been the focus of many theoretical studies on Neural Networks (NNs) [Neal, 1995, Poole et al., 2016, Schoenholz et al., 2017, Yang and Schoenholz, 2017, Hayou et al., 2019a, Lee et al., 2019]. Although unachievable in practice, it features many interesting properties which can help grasp the complex behaviour of large networks.

Infinitely wide 1-layer random NNs behave like Gaussian Processes (GPs) at initialization [Neal, 1995]. This was recently extended to multilayer NNs, where each layer can be associated to its own GP [Matthews et al., 2018, Lee et al., 2018, Yang, 2019a]. From a theoretical point of view, GPs have the advantage that their behaviour is fully captured by the mean function and the covariance kernel. Moreover, when dealing with GPs that are equivalent to infinite width NNs, these processes are usually centered, and hence fully determined by their covariance kernel. For multilayer networks, these kernels can be computed recursively, layer by layer [Lee et al., 2018]. Interestingly, in apparent contradiction with the naive idea "the deeper, the more expressive", it was shown in [Schoenholz et al., 2017] that the GP becomes trivial as the number of layers goes to infinity, that is the output completely forgets about the input and hence lacks expressive power. This loss of input information during the forward propagation through the network might be exponential in depth and could lead to trainability issues for extremely deep nets [Schoenholz et al., 2017, Hayou et al., 2019a].

One natural way to prevent this last issue is the introduction of skip connections, commonly known as the ResNet architecture. However, in the regime of large width and depth, the output of standard ResNets becomes inexpressive and the network may suffer from gradient exploding [Yang and Schoenholz, 2017].

In the present work, we propose a new class of residual neural networks, the Stable ResNet, which, in the limit of infinite width and depth, is shown to stabilize the gradient (no gradient vanishing or exploding) and to preserve expressivity in the limit of large depth. The main idea is the introduction of layer/depth dependent scaling factors to the ResNet blocks.

For ReLU networks, we provide a comprehensive analysis of two different scalings: a uniform one, where the scaling factor is the same for all the layers, and a decreasing one, where the scaling factor decreases as we go deeper inside the network. We also show that Stable ResNet solve the problem of Neural Tangent kernel (NTK) degeneracy in the limit of large depth [Hayou et al., 2019b]; indeed, with our scalings, the NTK is universal in the limit of infinite depth, which ensures that any continuous function can be approximated to an arbitrary precision by the features of the infinite depth NTK on a compact set.

All theoretical results are substantiated with numerical experiments in Section 7, where we demonstrate the benefits of Stable ResNet scalings both for the corresponding infinite width GP kernels as well as trained ResNets, over a range of moderate and large-scale image classification tasks: MNIST, CIFAR-10, CIFAR-100 and TinyImageNet.

*Equal contribution [1]Department of Statistics, University of Oxford. Correspondence to: <soufiane.hayou;eugenio.clerico;bobby.he@stats.ox.ac.uk>.

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. Copyright 2021 by the author(s).

2 RESNET

2.1 Setup and Notations

Consider a standard ResNet architecture with $L+1$ layers, labelled with $l \in [0:L]$[1], of dimensions $\{N_l\}_{l\in[0:L]}$.

$$\begin{aligned} &y_0(x) = W_0\, x + B_0\,; \\ &y_l(x) = y_{l-1}(x) + \mathcal{F}((W_l, B_l), y_{l-1}(x)) \quad \text{for } l \in [1:L]\,, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^d$ is an input, $y_l(x) = \{y_l^i(x)\}_{i\in[1:N_l]}$ is the vector of pre-activations, W_l and B_l are respectively the weights and bias of the l^{th} layer, and $\mathcal{F}$ is a mapping that defines the nature of the layer. In general, the mapping $\mathcal{F}$ consists of successive applications of simple linear maps (including convolutional layers), normalization layers [Ioffe and Szegedy, 2015] and activation functions. In this work, for the sake of simplicity, we consider Fully Connected blocks with ReLU activation function:

$$\mathcal{F}((W,B),x) = W\phi(x) + B\,,$$

where ϕ is the activation function. The weights and bias are initialized with $W_l \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_w^2/N_{l-1})$, and $B_l \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_b^2)$, where $\sigma_w > 0$, $\sigma_b \geq 0$, $N_{-1} = d$, and $\mathcal{N}(\mu, \sigma^2)$ is the normal law of mean μ and variance σ^2.

Recent results by [Hayou et al., 2021] suggest that scaling the residual blocks with $L^{-1/2}$ might have some beneficial properties on model pruning at initialization. This results from the stabilization effect on the gradient due to the scaling.

More generally, we introduce the residual architecture:

$$\begin{aligned} &y_0(x) = W_0\, x + B_0\,; \\ &y_l(x) = y_{l-1}(x) + \lambda_{l,L}\, \mathcal{F}((W_l, B_l), y_{l-1})\,, \quad l \in [1:L]\,, \end{aligned} \tag{2}$$

where $\{\lambda_{l,L}\}_{l\in[1:L]}$ is a sequence of scaling factors. We assume hereafter that there exists $\lambda_{\max} \in (0, \infty)$ such that $\lambda_{l,L} \in (0, \lambda_{\max}]$ for all $L \geq 1$ and $l \in [1:L]$.
In the next proposition, we give a necessary and sufficient condition for the gradient to remain bounded as the depth L goes to infinity.

Proposition 1 (Stable Gradient)**.** *Consider a ResNet of type (2), and let $\mathcal{L}_y(x) := \ell(y_L^1(x), y)$ for some $(x,y) \in \mathbb{R}^d \times \mathbb{R}$, where $\ell : (z,y) \mapsto \ell(z,y)$ is a loss function satisfying $\sup_{K_1\times K_2} \left|\frac{\partial \ell(z,y)}{\partial z}\right| < \infty$, for all compacts $K_1, K_2 \subset \mathbb{R}$. Then, in the limit of infinite width, for any compacts $K \subset \mathbb{R}^d$, $K' \subset \mathbb{R}$, there exists a constant $C > 0$ such that for all $(x,y) \in K \times K'$*

$$\sup_{l\in[0:L]} \mathbb{E}\left[\left|\frac{\partial \mathcal{L}_y(x)}{\partial W_l^{11}}\right|^2\right] \leq C \exp\left(\frac{\sigma_w^2}{2} \sum_{l=1}^L \lambda_{l,L}^2\right).$$

Moreover, if there exists $\lambda_{\min} > 0$ such that for all $L \geq 1$ and $l \in [1:L]$ we have $\lambda_{l,L} \geq \lambda_{\min}$, then, for all $(x,y) \in (\mathbb{R}^d \setminus \{0\}) \times \mathbb{R}$ such that $\left|\frac{\partial \ell(z,y)}{\partial z}\right| \neq 0$, there exists $\kappa > 0$ such that for all $l \in [1:L]$

$$\mathbb{E}\left[\left|\frac{\partial \mathcal{L}_y(x)}{\partial W_l^{11}}\right|^2\right] \geq \kappa \left(1 + \frac{\lambda_{\min}^2 \sigma_w^2}{2}\right)^L.$$

Proposition 1 shows that in order to stabilize the gradient, we have to scale the blocks of the ResNet with scalars $\{\lambda_{l,L}\}_{l\in[1:L]}$ such that $\sum_{l=1}^L \lambda_{l,L}^2$ remains bounded as the depth L goes to infinity. Taking $\lambda_{\min} = 1$, Proposition 1 shows that the standard ResNet architecture (1) suffers from gradient exploding at initialization,[2] which may cause instability during the first step of gradient based optimization algorithms such as Stochastic Gradient Descent (SGD). This motivates the following definition of Stable ResNet.

Definition 1 (Stable ResNet)**.** *A ResNet of type (2) is called a Stable ResNet if and only if $\lim_{L\to\infty} \sum_{l=1}^{L} \lambda_{l,L}^2 < \infty$.*

The condition on the scaling factors is satisfied by a wide range of sequences $\{\lambda_{l,L}\}_{l\in[1:L],L\geq 1}$. However, it is natural to consider the two categories:
Uniform scaling. The scaling factors have similar magnitude and tend to zero at the same time. A simple example is the uniform scaling $\lambda_{l,L} = 1/\sqrt{L}$.
Decreasing scaling. The sequence is decreasing and tends to zero. To be clearer, we consider a general sequence $\{\lambda_l\}_{l\in[1:L]}$ such that $\sum_{l\geq 1} \lambda_l^2 < \infty$, and let $\lambda_{l,L} = \lambda_l$ for all $L \geq 1$, all $l \in [1:L]$.

Note that our theoretical analyses will hold for any decreasing scaling $\{\lambda_l\}_{l\geq 1}$ that is square summable, but for simplicity in all empirical results we consider the decreasing scaling:

$$\lambda_l^{-1} = l^{1/2} \times \log(l+1)\,.$$

We study theoretical properties of both ResNets with uniform and decreasing scaling. We show that, in addition to stabilizing the gradient, both scalings ensure that the ResNet is expressive in the infinite depth limit. For this purpose, we use a tool known as Neural Network Gaussian Process (NNGP) [Lee et al., 2018] which is the equivalent Gaussian Process of a Neural Network in limit of infinite width.

2.2 On Gaussian Process approximation of Neural Networks

Consider a ResNet of type (2). Neurons $\{y_0^i(x)\}_{i\in[1:N_1]}$ are iid since the weights with which they are connected

[1] Notation: $[m:n] = \{m, m{+}1 \ldots n\}$ for integers $n \geq m$.

[2] In [Yang and Schoenholz, 2017], authors show a similar result with a slightly different ResNet architecture.

to the inputs are iid. Using the Central Limit Theorem, as $N_0 \to \infty$, $y_1^i(x)$ is a Gaussian variable for any input x and index $i \in [1 : N_1]$. Moreover, the variables $\{y_1^i(x)\}_{i\in[1:N_1]}$ are iid. Therefore, the processes $y_1^i(.)$ can be seen as independent (across i) centred Gaussian processes with covariance kernel Q_1. This is an idealized version of the true process corresponding to letting width $N_0 \to \infty$. Doing this recursively over l leads to similar approximations for $y_l^i(.)$ where $l \in [1 : L]$, and we write accordingly $y_l^i \overset{ind}{\sim} \mathcal{GP}(0, Q_l)$. The approximation of $y_l^i(.)$ by a Gaussian process was first proposed by [Neal, 1995] in the single layer case and was extended to multiple feedforward layers by [Lee et al., 2019] and [Matthews et al., 2018]. More recently, a powerful framework, known as Tensor Programs, was proposed by [Yang, 2019b], confirming the large-width NNGP association for nearly all NN architectures.

For any input $x \in \mathbb{R}^d$, we have $\mathbb{E}[y_l^i(x)] = 0$, so that the covariance $Q_l(x, x') = \mathbb{E}[y_l^1(x)y_l^1(x')]$ satisfies for all $x, x' \in \mathbb{R}^d$ (see Appendix A1)

$$Q_l(x, x') = Q_{l-1}(x, x') + \lambda_{l,L}^2 \Psi_{l-1}(x, x') \,,$$

where $\Psi_{l-1}(x, x') = \sigma_b^2 + \sigma_w^2 \mathbb{E}[\phi(y_{l-1}^1(x))\phi(y_{l-1}^1(x'))]$.

For the ReLU activation function $\phi : x \mapsto \max(0, x)$, the recurrence relation can be written more explicitly as in [Daniely et al., 2016]. Let C_l be the correlation kernel, defined as

$$C_l(x, x') = \frac{Q_l(x,x')}{\sqrt{Q_l(x,x)Q_l(x',x')}} \tag{3}$$

and let $f : [-1, 1] \to \mathbb{R}$ be given by

$$f : \gamma \mapsto \tfrac{1}{\pi}(\sqrt{1-\gamma^2} - \gamma \arccos \gamma) \,. \tag{4}$$

The recurrence relation reads (see Appendix A1)

$$\begin{aligned} Q_l &= Q_{l-1} + \lambda_{l,L}^2 \left[{\sigma_b}^2 + \frac{{\sigma_w}^2}{2}\left(1 + \frac{f(C_{l-1})}{C_{l-1}}\right) Q_{l-1}\right] \,, \\ Q_0(x, x') &= \sigma_b^2 + \sigma_w^2 \, \frac{x \cdot x'}{d} \,. \end{aligned} \tag{5}$$

This recursion leads to divergent diagonal terms $Q_L(x, x)$. This was proven in [Yang and Schoenholz, 2017] for a slightly different ResNet architecture. In the next Lemma, we extend this result to the ResNet defined by (1).

Lemma 1 (Exploding kernel with standard ResNet)**.** *Consider a ResNet of type* (1)*. Then, for all $x \in \mathbb{R}^d$,*

$$Q_L(x, x) \geq \left(1 + \frac{\sigma_w^2}{2}\right)^L \left(\sigma_b^2 \left(1 + \frac{2}{\sigma_w^2}\right) + \frac{\sigma_w^2}{d}\|x\|^2\right) .$$

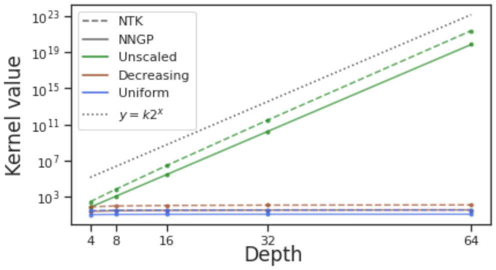

Figure 1: NNGP/NTK for unscaled ResNets explode exponentially (with base 2 if $\sigma_w^2 = 2$) in depth, unlike (both uniform and decreasing scaled) Stable ResNets.

Figure 1 plots the diagonal NNGP and NTK (introduced in Section 5) values for a point on the sphere, highlighting the exploding kernel problem for standard ResNets. Stable ResNets do not suffer from this problem.

We now introduce further notation and definitions. Hereafter, unless specified otherwise, K will denote a compact set in $\mathbb{R}^d$ ($d \geq 1$) and x, x' denote two arbitrary elements of K.
Let us start with a formal definition of a kernel.[3]

Definition 2 (Kernel)**.** *A kernel Q on K is a symmetric continuous function $K^2 \to \mathbb{R}$ such that, for all $n \in \mathbb{N}$, for any finite subset $\{x_1 \dots x_n\} \subset K$, the matrix $\{Q(x_i, x_j)\}_{i,j}$ is non-negative definite.*

The symmetry in the above definition has to be understood as $Q(x, x') = Q(x', x)$ for all $x, x' \in K$.

Kernels induce non-negative integral operators [Paulsen and Raghupathi, 2016].

Lemma 2. *Given a continuous and symmetric function $Q : K^2 \to \mathbb{R}$, we can define the induced integral operator $T(Q)$ on $L^2(K)$ via its action $T(Q)\varphi(x) = \int_K Q(x, y)\varphi(y)\,\mathrm{d}y$, for $\varphi \in L^2(K)$.*[4] *Moreover, $T(Q)$ is a bounded, compact, non-negative definite self-adjoint operator.*

Each kernel induces a centred Gaussian Process on K [Dudley, 2002], that is a random function F on K such that, for any finite $\hat{K} \subset K$, $\{F(x)\}_{x\in\hat{K}}$ is a centred Gaussian vector. We recall that the law of a centred GP is fully determined by its covariance function $(x, x') \mapsto \mathbb{E}[F(x)F(x')]$, defined on K^2.

Definition 3 (Induced GP)**.** *Given a kernel Q on K, the Gaussian Process induced by Q is a centred GP on K whose covariance function is Q.*

We will sometimes use the notation $\mathcal{GP}(0, Q)$ for the law of the GP induced by a kernel Q. With our definition

[3] Our definition is not the standard definition of a kernel, which is more general does not require the continuity, [Paulsen and Raghupathi, 2016].

[4] Naturally, we should write $L^2(K, \mu)$, specifying a measure μ on K. In the present work, unless otherwise specified, the notation $L^2(K)$ will imply the choice of any arbitrary finite Borel measure on K (cf Appendix A0).

of a kernel, the samples from the induced GP lies in $L^2(K)$ with probability 1 [Steinwart, 2019].

From now on we will assume that $0 \notin K$ if $\sigma_b = 0$.[5] For all ResNets, it is straightforward to check that Q_L is a kernel, in the sense of Definition 2 (see Appendix A1 or [Daniely et al., 2016]). The induced Gaussian Process is what we refer to as NNGP.

We denote by $\mathcal{H}_Q(K)$ the Reproducing Kernel Hilbert Space (RKHS)[6] induced by the kernel Q on the set K. The following hierarchical result holds.

Proposition 2. *For all $L \geq 1$, $l \in [0, L-1]$, $\mathcal{H}_{Q_l}(K) \subseteq \mathcal{H}_{Q_{l+1}}(K)$.*

Proposition 2 shows that, as we go deeper, the RKHS cannot become poorer. However, increasing L might introduce stability issues as illustrated in Proposition 1. We show in Sections 3 and 4 that Stable ResNets resolve this problem.

By Lemma 2, $T(Q_L)$ is a bounded, compact, self-adjoint operator and hence can be written as the sum of the projections on its eigenspaces [Lang, 2012]. By Mercer's Theorem [Paulsen and Raghupathi, 2016], all the eigenfunctions of $T(Q_L)$ are continuous. Finally, it is possible to link the eigen-decomposition of $T(Q_L)$ with the distribution of the GP induced by Q_L. Denoting respectively by μ_k and ψ_k the eigenvalues and eigenfunctions of the operator $T(Q_L)$, we have the equivalence in law:

$$y_L^1 \sim \sum_{k \in \mathbb{N}} \sqrt{\mu_k} Z_k \psi_k \sim \mathcal{GP}(0, Q_L)\,, \tag{6}$$

where $\{Z_k\}_{k\geq 0}$ are i.i.d. standard Gaussian random variables [Grenander, 1950]. The expressivity, that is the capacity to approximate a large class of function, of the network at initialization is then closely linked to the eigendecomposition of Q_L [Yang and Salman, 2019].

2.3 Universal kernels and expressive GPs

In this section, we provide a comprehensive study of the kernel Q_L. We start with a formal definition of universality (c-universality in [Sriperumbudur et al., 2011]). Again, unless otherwise stated, let K be a compact in $\mathbb{R}^d$.

Definition 4 (Universal Kernel)**.** *Let Q be a kernel on K, and $\mathcal{H}_Q(K)$ its RKHS [7]. We say that Q is universal on K if for any $\varepsilon > 0$ and any continuous function g on K, there exists $h \in \mathcal{H}_Q(K)$ such that $\|h - g\|_\infty < \varepsilon$.*

The universality of a kernel Q on a compact set implies that the kernel is strictly positive definite, i.e. for all non-zero $\varphi \in L^2(K), \langle T(Q)\varphi, \varphi\rangle > 0$ [Sriperumbudur et al., 2011]. Moreover, universality also implies the full expressivity of the induced GP, as expressed in the following.

Definition 5 (Expressive GP)**.** *A Gaussian Process on K is said to be expressive on $L^2(K)$ if, denoting by ψ a random realisation ψ of the process, for all $\varphi \in L^2(K)$, for all $\varepsilon > 0$,*

$$\mathbb{P}(\|\psi - \varphi\|_2 \leq \varepsilon) > 0\,.$$

Lemma 3. *A universal kernel Q on K induces an expressive GP on $L^2(K)$.*

By definition, universal kernels are characterized by the property that their associated RKHS is dense (w.r.t the uniform norm $\|.\|_\infty$) in the space of continuous functions on K. This is crucial for Kernel regression and Gaussian Process inference [Kanagawa et al., 2018].[8] By Proposition 2, it suffices to prove that Q_{L_0} is universal for some L_0 in order to conclude for all $L \geq L_0$. It turns out this is true for $L_0 = 2$.

Proposition 3. *If $\sigma_b > 0$, then Q_2 is universal on K. From Proposition 2, Q_L is universal for all $L \geq 2$.*

Note that the presence of biases is essential to achieve universality in the case of a general K, since the output of a ReLU ResNet with no bias is always a positive homogeneous function of its input, i.e., a map F such that $F(\alpha x) = \alpha F(x)$ for all $\alpha \geq 0$. However, in the particular case of $K = \mathbb{S}^{d-1}$, the unit sphere in $\mathbb{R}^d$, the kernel Q_L is universal (for $L \geq 2$), even when $\sigma_b = 0$.

Proposition 4. *Assume $\sigma_b = 0$. Then for all $L \geq 2$, Q_L is universal on $\mathbb{S}^{d-1}$ for $d \geq 2$.*

Another interesting fact of the case $K = \mathbb{S}^{d-1}$ is that the eigendecomposition of the kernel Q_L has a simple structure. Indeed, on $\mathbb{S}^{d-1}$, $Q_L(x, x')$ depends only on the scalar product $x \cdot x'$. These kernels (zonal kernel) admit Spherical Harmonics as an eigenbasis [Yang and Salman, 2019].

Proposition 5 (Spectral decomposition on $\mathbb{S}^{d-1}$)**.** *Let Q be a zonal kernel on $\mathbb{S}^{d-1}$, that is $Q(x, x') = p(x \cdot x')$ for a continuous function $p : [-1, 1] \to \mathbb{R}$. Then, there is a sequence $\{\mu_k \geq 0\}_{k \in \mathbb{N}}$ such that for all $x, x' \in \mathbb{S}^{d-1}$*

$$Q(x, x') = \sum_{k \geq 0} \mu_k \sum_{j=1}^{N(d,k)} Y_{k,j}(x) Y_{k,j}(x')\,,$$

where $\{Y_{k,j}\}_{k\geq 0, j \in [1:N(d,k)]}$ are spherical harmonics of $\mathbb{S}^{d-1}$ and $N(d, k)$ is the number of harmonics of order

[5] We exclude 0 since for $\sigma_b = 0$ C_0 is discontinuous in 0 and can't be a kernel on K as in Definition 2, if $0 \in K$.

[6] See Appendix A0 for a definition.

[7] See Appendix A0.

[8] The closure of the set of functions described by the mean function of the posterior of a GP regression is exactly the RKHS of the kernel of the GP prior.

k. With respect to the standard spherical measure, the spherical harmonics form an orthonormal basis of $L^2(\mathbb{S}^{d-1})$ and $T(Q)$ is diagonal on this basis.

Although the kernel is universal for fixed depth L, it is not guaranteed that as $L \to \infty$, Q_L remains universal. Indeed, for the standard ResNet architecture, the variance $Q_L(x,x)$ grows exponentially with L [Yang and Schoenholz, 2017], and therefore, the kernel diverges. In order to analyse the expressivity of the kernel of a standard ResNet in the limit of large depth, we can study the correlation kernel C_L, defined in (3), instead. We show in the following Lemma that, as L goes to infinity, the kernel C_L converges to a constant (which has a 1D RKHS).

Lemma 4. *Consider a standard ResNet of type (1) and let $K \subset \mathbb{R}^d \setminus \{0\}$ be a compact set. We have that*

$$\lim_{L\to\infty} \sup_{x,x'\in K} |1 - C_L(x,x')| = 0\,.$$

Moreover, if $\sigma_b = 0$, then,

$$\sup_{x,x'\in K} |1 - C_L(x,x')| = \mathcal{O}(L^{-2})\,.$$

Therefore, $\mathcal{H}_{C_\infty}(K)$ is the space of constant functions.

Lemma 4 shows that in the limit of infinite depth L, the RKHS of the correlation kernel is trivial, meaning that the NNGP cannot be expressive. On the contrary, we will show in the next sections that Stable ResNets achieve a universal kernel for infinite depth L.

3 UNIFORM SCALING

Consider a Stable ResNet with layers $[0 : L]$. Under uniform scaling, the recurrence relation in (5) reads:

$$Q_l = Q_{l-1} + \tfrac{1}{L}\left({\sigma_b}^2 + \tfrac{{\sigma_w}^2}{2}\left(1 + \tfrac{f(C_{l-1})}{C_{l-1}}\right) Q_{l-1}\right). \quad (7)$$

In the limit as $L \to \infty$, (7) converges uniformly to a continuous ODE. Studying the solution of the corresponding Cauchy problem, we show that the covariance kernel remains universal in the limit of infinite depth.

3.1 Continuous formulation

The layer index l in (7) can be rescaled as $l \mapsto t(l) = l/L$. Clearly $t(0) = 0$ and $t(L) = 1$, so the image of t is contained in $[0, 1]$. In the limit $L \to \infty$ it is natural to consider t as a continuous variable spanning the interval $[0, 1]$. With this in mind, it makes sense to look at the continuous version of (7).
Let $K \subset \mathbb{R}^d$ be a compact set and $x, x' \in K$. If $\sigma_b = 0$ assume that $0 \notin K$.

$$\begin{aligned} \dot{q}_t(x,x') &= {\sigma_b}^2 + \tfrac{{\sigma_w}^2}{2}\left(1 + \tfrac{f(c_t(x,x'))}{c_t(x,x')}\right) q_t(x,x')\,, \\ q_0(x,x') &= \sigma_b^2 + \sigma_w^2\, \tfrac{x\cdot x'}{d}\,, \\ c_t(x,x') &= \tfrac{q_t(x,x')}{\sqrt{q_t(x,x)q_t(x',x')}}\,. \end{aligned} \quad (8)$$

As discussed in Section A2 of the Appendix, for any x, x', the solution of the above Cauchy problem exists and is unique. Moreover, the solutions q_t and c_t are kernels on K, in the sense of Definition 2.

Clearly, for finite L, the continuous ODE (8) is an approximation. However, the following result holds.

Lemma 5 (Convergence to the continuous limit)**.** *Let $Q_{l|L}$ be the covariance kernel of the layer l in a net of $L+1$ layers $[0 : L]$, and q_t be the solution of (8), then*

$$\lim_{L\to\infty} \sup_{l\in[0:L]} \sup_{(x,x')\in K^2} |Q_{l|L}(x,x') - q_{t=l/L}(x,x')| = 0\,.$$

3.2 Universality of the covariance kernel

When $\sigma_b > 0$, the kernel q_t is universal for $t > 0$.

Theorem 1 (Universality of q_t)**.** *Let $K \subset \mathbb{R}^d$ be compact and assume $\sigma_b > 0$. For any $t \in (0, 1]$, the solution q_t of (8) is a universal kernel on K.*

The proof of the above statement is detailed in Appendix A2. The main idea is to show that the integral operator $T(q_t)$ is strictly positive definite and then use a characterization of universal kernels, due to [Sriperumbudur et al., 2011], which connects the universality of Definition 4 with the strict positivity of the induced integral operator.[9]

As mentioned previously, the presence of the bias is essential to achieve full expressivity on a generic compact $K \subset \mathbb{R}^d$. However, we can still have universality when no bias is present, limiting ourselves to the case of the unit sphere $K = \mathbb{S}^{d-1}$.

Proposition 6 (Universality on $\mathbb{S}^{d-1}$)**.** *For any $t \in (0, 1]$, the covariance kernel q_t, solution of (8) with $\sigma_b = 0$, is universal on $\mathbb{S}^{d-1}$, with $d \geq 2$.*

4 DECREASING SCALING

Consider a Stable ResNet with decreasing scaling, that is a sequence of scaling factors $(\lambda_k)_{k\geq1}$ such that $\sum_{k\geq1} \lambda_k^2 < \infty$. In this setting, each additional layer can be seen as a correction to the network output with decreasing magnitude. As for the uniform scaling, we

[9] The details are more involved as we need to show that the kernel induces a strictly positive definite operator on $L^2(K, \mu)$ for any finite Borel measure μ on K.

show in the next proposition that the kernel Q_L converges to a limiting kernel Q_∞, and the convergence is uniform over any compact set of $\mathbb{R}^d$. The notation $g(x) = \Theta(m(x))$ means there exist two constants $A, B > 0$ such that $Am(x) \leq g(x) \leq Bm(x)$.

Proposition 7 (Uniform Convergence of the Kernel). *Consider a Stable ResNet with a decreasing scaling, i.e. the sequence $\{\lambda_l\}_{l\geq 1}$ is such that $\sum_l \lambda_l^2 < \infty$. Then for all $(\sigma_b, \sigma_w) \in \mathbb{R}^+ \times (\mathbb{R}^+)^*$, there exists a kernel Q_∞ on $\mathbb{R}^d$ such that for any compact set $K \subset \mathbb{R}^d$,*

$$\sup_{x,x' \in K} |Q_L(x,x') - Q_\infty(x,x')| = \Theta\big(\textstyle\sum_{k\geq L} \lambda_k^2\big)\,.$$

The convergence of the kernel Q_L to the limiting kernel Q_∞ is governed by the convergence rate of the series of scaling factors. Moreover, leveraging the RKHS hierarchy from Proposition 2, we find that Q_∞ is universal.

Corollary 1 (Universality of Q_∞). *The following statements hold*

- *Let K be a compact set of $\mathbb{R}^d$ and assume $\sigma_b > 0$. Then, Q_∞ is universal on K.*
- *Assume $\sigma_b = 0$. Then Q_∞ is universal on $\mathbb{S}^{d-1}$.*

As in the uniform scaling case, the limiting kernel exists and is universal unlike the standard ResNet architecture that yields a divergent kernel Q_L as $L \to \infty$.

To validate our universality and expressivity results, Figure 2 plots the leading eigenvalues of the NNGP (& NTK, introduced in Section 5) kernels on a set of 1000 points sampled uniformly at random from the circle, normalized so that the largest eigenvalue is 1. We use the recursion formulas for NNGP correlation (Lemma A4) and normalized NTK (Lemma A19) to avoid the exploding variance/gradient problem. We see that the unscaled ResNet NNGP becomes inexpressive with depth because all non-leading eigenvalues converge to 0, whereas our Stable ResNets (decreasing and uniform scaling) are expressive even in the large depth limit.

5 NEURAL TANGENT KERNEL

In the so-called lazy training regime [Chizat and Bach, 2019], the training dynamics of an infinitely wide network can be described via the Neural Tangent Kernel (NTK) [Lee et al., 2019], introduced in [Jacot et al., 2018] and defined as

$$\tilde{\Theta}_L^{ij}(x,x') = \nabla_{\text{par}}\, y_L^i(x) \cdot \nabla_{\text{par}}\, y_L^j(x')\,,$$

with ∇_{par} the gradient wrt the parameters of the NN.[10] To simplify our presentation we will assume that the output dimension of the network is 1.[11]

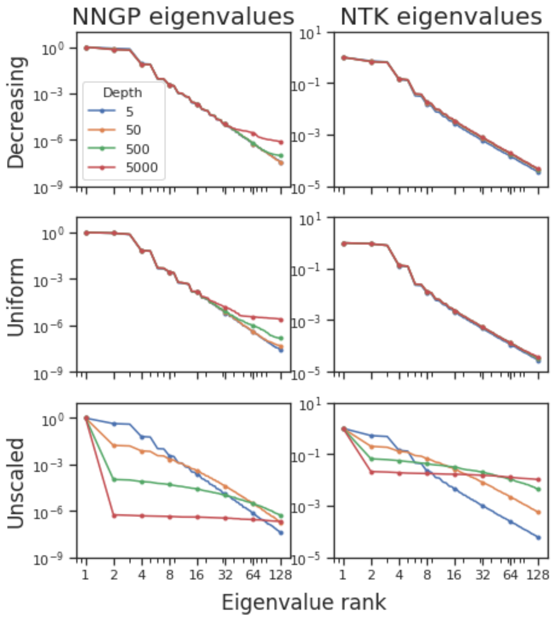

Figure 2: (Normalized) NNGP & NTK matrix eigenvalues of Stable (decreasing & uniform) & unscaled (i.e. standard) ResNets.

Let F_τ be the output function of the ResNet at training time τ. In the NTK regime (infinite width), the gradient flow is equivalent to a simple linear model [Lee et al., 2019], that gives

$$F_\tau(x) - F_0(x) = \Theta_L(x,\mathcal{X})\hat{\Theta}_L^{-1}(I - e^{-\eta\hat{\Theta}_L\tau})(\mathcal{Y} - F_0(\mathcal{X}))\,,$$

where $\mathcal{X}$ and $\mathcal{Y}$ are respectively the input and output datasets, $\Theta_L(x,\mathcal{X}) = \{\Theta_L(x,x')\}_{x'\in\mathcal{X}}$ and $\hat{\Theta}_L$ is the matrix $\{\Theta_l(x,x')\}_{x,x'\in\mathcal{X}}$. The universality of the NTK is crucial for the ResNet to learn beyond initialization, since the residual $F_\tau - F_0$ lies in the RKHS generated by Θ_L. For unscaled ResNet, [Hayou et al., 2019b] showed that the limiting NTK is trivial in the sense of Lemma 4. However, this is not the case for Stable ResNet.

Consider a ResNet of type (2). We have [12]

$$\Theta_0 = Q_0,\ \ \Theta_{l+1} = \Theta_l + \lambda_{l,L}^2\left(\Psi_l + \Psi_l'\,\Theta_l\right)\,, \qquad (9)$$

where $\Psi_l(x,x') = \sigma_b^2 + \sigma_w^2\mathbb{E}[\phi(y_1^l(x))\phi(y_1^l(x'))]$ and $\Psi_l'(x,x') = \sigma_w^2\mathbb{E}[\phi'(y_1^l(x))\phi'(y_1^l(x'))]$ (see Appendix A4).

Proposition 8. *Fix a compact $K \subset \mathbb{R}^d$ ($0 \notin K$ if $\sigma_b = 0$) and consider a Stable ResNet with decreasing scaling. Then Θ_L converges uniformly over K^2 to a kernel Θ_∞. Moreover Θ_∞ is universal on K if $\sigma_b > 0$. If $K = \mathbb{S}^{d-1}$, then the universality holds for $\sigma_b = 0$.*

[10] All network considered in this section are assumed to have NTK parametrization, cf Appendix A4 for details.

[11] This does not affect our final conclusion of universality for the NTK, which is diagonal in the output space, that is $\tilde{\Theta}^{ij} = \Theta\delta^{ij}$, [Jacot et al., 2018, Hayou et al., 2019b].

[12] This is true under the technical assumption that the parameters appearing in the back-propagation can be considered independent from the ones of the forward pass (Gradient Independent Assumption) [Yang, 2019a]

An analogous result can be stated for the uniform scaling, after noticing that a continuous formulation ($\Theta_l \mapsto \theta_{t(l)}$) can be obtained in analogy with what has been done for the covariance kernel (cf Appendix A4).

Proposition 9. *Let $K \subset \mathbb{R}^d$ and fix $t \in (0,1]$. If $\sigma_b > 0$, then θ_t is universal on K. The same holds true if $\sigma_b = 0$ and $K = \mathbb{S}^{d-1}$.*

Figure 2 shows that the non-leading NTK eigenvalues do not decay to 0 with depth for Stable ResNets, unlike for unscaled ResNets. This is in line with findings of Propositions 8 and 9.

6 A PAC-BAYES RESULT

Consider a dataset S with N iid training examples $(x_i, y_i)_{1\leq i \leq N} \in X \times Y$, and a hypothesis space $\mathcal{P}$ from which we want to learn an optimal hypothesis according to some bounded loss function $\ell : Y \times Y \mapsto [0,1]$. The empirical/generalization loss of a hypothesis $h \in \mathcal{U}$ are

$$r_S(h) = \tfrac{1}{N}\textstyle\sum_{i=1}^N \ell(h(x_i), y_i), \quad r(h) = \mathbb{E}_\nu[\ell(h(x), y)]\,,$$

where ν is a probability distribution on $X \times Y$. For some randomized learning algorithm $\mathcal{A}$, the empirical and generalization loss are given by:

$$r_S(\mathcal{A}) = \mathbb{E}_{h\sim\mathcal{A}}[r_S(h)], \qquad r(\mathcal{A}) = \mathbb{E}_{h\sim\mathcal{A}}[r(h)]\,.$$

The PAC-Bayes theorem gives a probabilistic upper bound on the generalization loss $r(\mathcal{A})$ of a randomized learning algorithm $\mathcal{A}$ in terms of the empirical loss $r_S(\mathcal{A})$. Fix a prior distribution $\mathcal{P}$ on the hypothesis set $\mathcal{U}$. The Kullback-Leibler divergence between $\mathcal{A}$ and $\mathcal{P}$ is defined as $\mathrm{KL}(\mathcal{A}\|\mathcal{P}) = \int \mathcal{A}(h)\log\frac{\mathcal{A}(h)}{\mathcal{P}(h)}\mathrm{d}h \in [0,\infty]$. The Bernoulli KL-divergence is given by $\mathrm{kl}(a||p) = a\log\frac{a}{p} + (1-a)\log\frac{1-a}{1-p}$ for $a, p \in [0,1]$. We define the inverse Bernoulli KL-divergence kl^{-1} by

$$\mathrm{kl}^{-1}(a,\varepsilon) = \sup\{p \in [0,1] : \mathrm{kl}(a||p) \leq \varepsilon\}\,.$$

Theorem 2 (PAC-Bayes bound Theorem [Seeger, 2002]). *For any loss function ℓ that is $[0,1]$ valued, any distribution ν, any $N \in \mathbb{N}$, any prior $\mathcal{P}$, and any $\delta \in (0,1]$, with probability at least $1-\delta$ over the sample S, we have*

$$\forall\mathcal{A}, \quad r(\mathcal{A}) \leq \mathrm{kl}^{-1}\left(r_S(\mathcal{A}), \tfrac{\mathrm{KL}(\mathcal{A}\|\mathcal{P}) + \log(2\sqrt{N}/\delta)}{N}\right)\,.$$

The KL-divergence term $\mathrm{KL}(\mathcal{A}\|\mathcal{P})$ plays a major role as it controls the generalization gap, i.e. the difference (in terms of Bernoulli KL-divergence) between the empirical loss and the generalization loss. In our setting, we consider an ordinary GP regression with prior $\mathcal{P}(f) = \mathcal{GP}(f|0, Q(x,x'))$. Under the standard assumption that the outputs $y_N = (y_i)_{i\in[1:N]}$ are noisy versions of $f_N = (f(x_i))_{i\in[1:N]}$ with $y_N|f_N \sim \mathcal{N}(y_N|f_N, \sigma^2 I)$, the Bayesian posterior $\mathcal{A}$ is also a GP and is given by

$$\begin{aligned}\mathcal{A}(f) = \mathcal{GP}(f|Q_N(x)(Q_{NN} + \sigma^2 I)^{-1}y_N, Q(x,x')\\ - Q_N(x)(Q_{NN}+\sigma^2 I)^{-1})Q_N(x')^T)\,.\end{aligned} \tag{10}$$

$Q_N(x) = (Q(x,x_i))_{i\in[1:N]}$, $Q_{NN} = (Q(x_i,x_j))_{1\leq i,j\leq N}$. In this setting, we have the following result

Proposition 10 (Curse of Depth)**.** *Let Q_L be the kernel of a ResNet. Let P_L be a GP with kernel Q_L and $\mathcal{A}_L$ be the corresponding Bayesian posterior for some fixed noise level $\sigma^2 > 0$. Then, in a fixed setting (fixed sample size N), the following results hold:*

- *With a standard ResNet, $\mathrm{KL}(\mathcal{A}_L\|P_L) \gtrsim L$.*
- *With a Stable ResNet, $\mathrm{KL}(\mathcal{A}_L\|P_L) = \mathcal{O}_L(1)$.*

The KL-divergence bound diverges for a standard ResNet while it remains bounded for Stable ResNet. Although PAC-Bayes bounds only give an upper bound on the generalization error, Proposition 10 shows that Stable ResNet does not suffer from the "curse of depth", i.e. the KL-divergence does not explode as the depth becomes large.

7 EXPERIMENTS

In line with our theory, we now present results demonstrating empirical advantages of Stable ResNets (both uniform and decreasing scaling) compared to their unscaled counterparts on a toy regression task and standard image classification tasks, both for infinite-width NNGP kernels as well as trained finite-width NNs in the latter case. In the interests of space, all experimental details not described in this section can be found in Appendix A7. All error bars in this section correspond to 3 independent runs.

Stable NNGP regression experiment We first present a toy regression posterior regression experiment with NNGP kernel. We compare across different depths and scalings, with target test function $y = x\sin(x)$ and a small amount of observation noise $\sigma = 0.1$ (σ as defined in Eq. 10). We use 5 training points (dark green dots).

We map our 1D inputs x onto the circle $(\cos(x), \sin(x))$ before performing GP regression. This is so that all inputs have unit norm and we can use the NNGP correlation kernel (Eq. 3) for the vanilla ResNet (ResNet with fully connected blocks), in order to avoid the exploding variance problem.

As expected from our theory, in Figure 3, for depth 1000 the NNGP correlation kernel without stable scaling (top row, red) is unable to learn anything beyond a constant function due to inexpressivity, whereas our

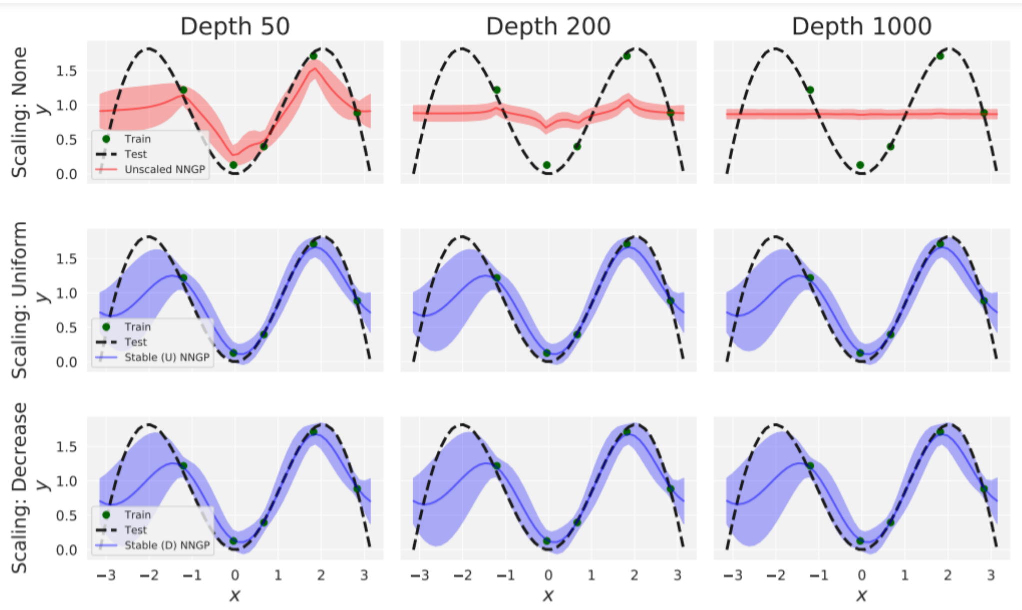

Figure 3: NNGP toy regression experiment. Stable ResNets (bottom two rows, blue) are still expressive in the large depth limit. We plot mean and 95% posterior predictive credible interval for NNGP posteriors.

Stable NNGP classification results We first compare the performance of Stable and standard ResNets of varying depths through their infinite-width NNGP kernels, on MNIST & CIFAR-10. For each considered NNGP kernel Q and training set $(x_i, y_i)_{i\in[1:N]}$, we report test accuracy using the mean of the posterior predictive (Eq. 10): $Q_N(\cdot)(Q_{NN} + \sigma^2 I)^{-1} y_N$, which is also the kernel ridge regression predictor [Kanagawa et al., 2018]. We treat classification labels y as one-hot regression targets, similar to recent works [Arora et al., 2019, Lee et al., 2019, Shankar et al., 2020], and tune the noise σ^2 using prediction accuracy on a held-out validation set.

Table 1: CIFAR-10 test accuracies (%) using posterior predictive mean of NNGP kernels for deep Wide-ResNets [Zagoruyko and Komodakis, 2016] with different training set sizes N. Scaled (D) & Scaled (U) refer to decreasing and uniform scaling respectively.

N	Depth	Scaled (D)	Scaled (U)	Unscaled
1K	112	$36.84_{\pm 0.53}$	$36.43_{\pm 0.49}$	$37.71_{\pm 0.50}$
	202	$36.89_{\pm 0.55}$	$36.47_{\pm 0.49}$	—
10K	112	$53.81_{\pm 0.11}$	$53.55_{\pm 0.41}$	$53.34_{\pm 0.07}$
	202	$53.80_{\pm 0.10}$	$53.57_{\pm 0.40}$	—

First, in Table 1, we demonstrate the exploding NNGP variance problem for unscaled Wide-ResNets (WRN) [Zagoruyko and Komodakis, 2016]. For an unscaled WRN of depth 202, the NNGP kernel values explode resulting in numerical errors, whereas Stable ResNets achieve 54% test accuracy with 10K training points (out of full size 50K). Note that any numerical errors from exploding NNGP also afflict the NTK, as the difference between the NTK and NNGP is positive semi-definite [Lee et al., 2019, He et al., 2020] (which is why the NTK lines always lie above their corresponding NNGP in Figure 1).

To isolate the disadvantages of inexpressivity in unscaled Resnets NNGPs compared to our Stable ResNets, we need to avoid the exploding variance problem and ensuing numerical errors. In order to do so, we use the NNGP correlation kernel C instead of the NNGP covariance kernel Q, noting that these two kernels are equal up to multiplicative constant on the sphere, and that the posterior predictive mean is invariant to the scale of Q (with σ^2 also tuned relative to the scale of Q). Moreover, the formula in Lemma A4 for NNGP correlation recursion for vanilla ResNets without bias can be recast as a ResNet with a modified scaling (see Appendix A6), allowing us to use existing optimised libraries [Novak et al., 2020]. In order to use the vanilla ResNet correlation recursion, we standardise all MNIST & CIFAR-10 images to lie on the 784 & 3072-dimension sphere respectively.

Our expressivity results, as well as Proposition 10, suggest that we expect Stable ResNets to outperform standard ResNets for large depths even when exploding variance numerical errors are alleviated for standard ResNets. In Table 2, we see that unscaled ResNets suffer from a degradation in test accuracy with depth, due to inexpressivity, whereas our Stable ResNets (both de-

Table 2: MNIST and CIFAR-10 test accuracies (%) using posterior predictive mean of NNGP kernels for deep vanilla ResNets (ResNet with fully connected blocks) with different size training sets N.

	Dataset	MNIST			CIFAR-10		
N	Depth	Scaled (D)	Scaled (U)	Unscaled	Scaled (D)	Scaled (U)	Unscaled
1K	50	92.88$_{\pm 0.35}$	92.39$_{\pm 0.33}$	92.44$_{\pm 0.21}$	35.83$_{\pm 0.14}$	34.73$_{\pm 0.14}$	**37.16**$_{\pm 0.25}$
	200	92.91$_{\pm 0.35}$	92.39$_{\pm 0.32}$	89.56$_{\pm 0.56}$	**35.86**$_{\pm 0.14}$	34.76$_{\pm 0.11}$	34.85$_{\pm 0.17}$
	1000	92.92$_{\pm 0.34}$	92.39$_{\pm 0.32}$	55.13$_{\pm 5.31}$	**35.89**$_{\pm 0.14}$	34.76$_{\pm 0.11}$	12.43$_{\pm 3.97}$
10K	50	97.57$_{\pm 0.12}$	97.55$_{\pm 0.12}$	97.06$_{\pm 0.10}$	48.71$_{\pm 0.31}$	48.12$_{\pm 0.27}$	**50.11**$_{\pm 0.37}$
	200	97.57$_{\pm 0.11}$	97.55$_{\pm 0.12}$	95.55$_{\pm 0.13}$	**48.77**$_{\pm 0.30}$	47.15$_{\pm 0.18}$	47.00$_{\pm 0.30}$
	1000	97.57$_{\pm 0.10}$	97.54$_{\pm 0.12}$	67.53$_{\pm 2.96}$	**48.76**$_{\pm 0.30}$	47.16$_{\pm 0.17}$	17.86$_{\pm 2.32}$

creasing and uniform) do not suffer from a drop in performance. For example, the posterior predictive mean using the NNGP of an unscaled vanilla ResNet with depth 1000 attains only 17.86% accuracy on CIFAR-10 with 10K training points, compared to 48.76% for Stable ResNet (decreasing scale).

We focus on the NNGP rather than the NTK as recent works [Lee et al., 2020, Shankar et al., 2020] have demonstrated that there is no advantage to the state-of-the-art NTK over the NNGP as infinite-width kernel predictors. Moreover, we do not aim for near state-of-the-art kernel results due to computational resources, and instead aim to empirically validate the theoretical advantages of Stable ResNets.

Trained Stable ResNet results Finally, we consider the benefits of trained Stable ResNets on the large-scale CIFAR-10, CIFAR-100 and TinyImageNet[13] datasets. We compare trained convolutional ResNets [He et al., 2016] of depths 32, 50 & 104 in terms of test accuracy. In the main text we present results for ResNets trained with Batch Normalization [Ioffe and Szegedy, 2015] (BatchNorm), while results for trained ResNets without BatchNorm can be found in Appendix A7. Stable ResNet scalings are applied to the residual connection after all convolution, ReLU and BatchNorm layers.

We use initial learning rate 0.1 which is decayed by 0.1 at 50% and 75% of the way through training. This learning rate schedule has been used previously [He et al., 2016] for unscaled ResNets and we found it to work well for all ResNets trained with BatchNorm. We train for 160 epochs on CIFAR-10/100 and 250 epochs on TinyImageNet. Test accuracy results are displayed in Table 3. As we can see, Stable ResNets consistently outperform standard ResNets across datasets and depths. Moreover, the performance gap is larger for larger depths: for example on CIFAR-100 our Stable ResNet (decreasing) outperforms its standard counterpart by 1.05% (75.06 vs 74.01) on average for depth 32 whereas for depth 104 the test accuracy gap is 2.36% (77.44 vs 75.08) on average. A similar trend can also be observed for the more challenging TinyImageNet dataset. Interestingly, we see that among the Stable ResNets, decreasing scaling also consistently outperforms uniform scaling.

Table 3: Test accurracies (%) of trained deep ResNets of various scalings and depths on CIFAR-10 (C-10), CIFAR-100 (C-100) & TinyImageNet (Tiny-I).

Dataset	Depth	Scaled (D)	Scaled (U)	Unscaled
C-10	32	94.84$_{\pm 0.08}$	94.78$_{\pm 0.17}$	94.66$_{\pm 0.07}$
	50	**95.07**$_{\pm 0.06}$	94.99$_{\pm 0.03}$	94.85$_{\pm 0.06}$
	104	95.14$_{\pm 0.19}$	**95.31**$_{\pm 0.07}$	95.10$_{\pm 0.21}$
C-100	32	**75.06**$_{\pm 0.05}$	74.79$_{\pm 0.28}$	74.01$_{\pm 0.14}$
	50	**76.20**$_{\pm 0.22}$	75.81$_{\pm 0.20}$	74.66$_{\pm 0.33}$
	104	**77.44**$_{\pm 0.09}$	76.88$_{\pm 0.39}$	75.08$_{\pm 0.42}$
Tiny-I	32	63.01$_{\pm 0.22}$	**63.06**$_{\pm 0.04}$	62.79$_{\pm 0.08}$
	50	64.78$_{\pm 0.24}$	64.74$_{\pm 0.10}$	63.96$_{\pm 0.39}$
	104	66.57$_{\pm 0.39}$	66.67$_{\pm 0.12}$	65.27$_{\pm 0.52}$

8 CONCLUSION

Stable ResNets have the benefit of stabilizing the gradient and ensuring expressivity in the limit of infinite depth. We have demonstrated theoretically and empirically that this type of scaling makes NNGP inference robust and improves test accuracy with SGD on modern ResNet architectures. However, while Stable ResNets with both uniform and decreasing scalings outperform standard ResNet, the selection of an optimal scaling remains an open question; we leave this topic for future work.

[13] Available at http://cs231n.stanford.edu/tiny-imagenet-200.zip

ACKNOWLEDGMENTS

This material is based upon work supported in part by the U.S. Army Research Laboratory and the U. S. Army Research Office, and by the U.K. Ministry of Defence (MoD) and the U.K. Engineering and Physical Research Council (EPSRC) under grant number EP/R013616/1. AD is also partially supported by EPSRC EP/R034710/1. BH is supported by the EPSRC and MRC through the OxWaSP CDT programme (EP/L016710/1). The project leading to this work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 834175).

References

G. Yang and S. Schoenholz. Mean field residual networks: On the edge of chaos. In *Advances in Neural Information Processing Systems*, pages 7103–7114, 2017.

S. Hayou, A. Doucet, and J. Rousseau. On the impact of the activation function on deep neural networks training. In *International Conference on Machine Learning*, 2019a.

R.M. Neal. *Bayesian Learning for Neural Networks*, volume 118. Springer Science & Business Media, 1995.

B. Poole, S. Lahiri, M. Raghu, J. Sohl-Dickstein, and S. Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *Advances in Neural Information Processing Systems*, 2016.

S.S. Schoenholz, J. Gilmer, S. Ganguli, and J. Sohl-Dickstein. Deep information propagation. In *International Conference on Learning Representations*, 2017.

J. Lee, L. Xiao, S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In *Advances in Neural Information Processing Systems*. 2019.

A.G. Matthews, J. Hron, M. Rowland, R.E. Turner, and Z. Ghahramani. Gaussian process behaviour in wide deep neural networks. In *International Conference on Learning Representations*, 2018.

J. Lee, Y. Bahri, R. Novak, S.S. Schoenholz, J. Pennington, and J. Sohl-Dickstein. Deep neural networks as Gaussian processes. In *International Conference on Learning Representations*, 2018.

G. Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *arXiv preprint arXiv:1902.04760*, 2019a.

S. Hayou, A. Doucet, and J. Rousseau. Mean-field behaviour of neural tangent kernel for deep neural networks. *arXiv preprint arXiv:1905.13654*, 2019b.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*, 2015.

S. Hayou, J.F. Ton, A. Doucet, and Y.W. Teh. Robust pruning at initialization. In *International Conference on Learning Representations*, 2021.

G. Yang. Tensor programs i: Wide feedforward or recurrent neural networks of any architecture are Gaussian processes. *arXiv preprint arXiv:1910.12478*, 2019b.

A. Daniely, R. Frostig, and Y. Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. *Advances in Neural Information Processing Systems 29*, 2016.

V.I. Paulsen and M. Raghupathi. *An Introduction to the Theory of Reproducing Kernel Hilbert Spaces*. Cambridge University Press, 2016.

R. M. Dudley. *Real Analysis and Probability*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2 edition, 2002.

I. Steinwart. Convergence types and rates in generic Karhunen-Loeve expansions with applications to sample path properties. *Potential Analysis*, 51(3): 361–395, 2019.

S. Lang. *Real and Functional Analysis*. Graduate Texts in Mathematics. Springer, New York, 3rd edition, 2012.

U. Grenander. Stochastic processes and statistical inference. *Arkiv Matematik*, 1(3):195–277, 10 1950.

G. Yang and H. Salman. A fine-grained spectral perspective on neural networks. *arXiv preprint 1907.10599*, 2019.

B. Sriperumbudur, K. Fukumizu, and G. Lanckriet. Universality, characteristic kernels and RKHS embedding of measures. *Journal of Machine Learning Research*, 12(70):2389–2410, 2011.

M. Kanagawa, P. Hennig, D. Sejdinovic, and B.K. Sriperumbudur. Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*, 2018.

L. Chizat and F. Bach. A note on lazy training in supervised differentiable programming. In *Advances in Neural Information Processing Systems*, 2019.

A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, 2018.

M. Seeger. PAC-Bayesian generalisation error bounds for Gaussian process classification. *Journal of Machine Learning Research*, page 233–269, 02 2002.

S. Arora, S.S. Du, W. Hu, Z. Li, R. Salakhutdinov, and R. Wang. On exact computation with an infinitely wide neural net. In *Advances in Neural Information Processing Systems*, 2019.

V. Shankar, A. Fang, W. Guo, S. Fridovich-Keil, L. Schmidt, J. Ragan-Kelley, and B. Recht. Neural kernels without tangents. *International Conference on Machine Learning*, 2020.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference 2016*, 2016.

B. He, B. Lakshminarayanan, and Y. W. Teh. Bayesian deep ensembles via the neural tangent kernel. *Advances in Neural Information Processing Systems*, 2020.

R. Novak, L. Xiao, J. Hron, J. Lee, A. Alemi, J. Sohl-Dickstein, and S. Schoenholz. Neural tangents: Fast and easy infinite neural networks in Python. In *International Conference on Learning Representations*, 2020. URL https://github.com/google/neural-tangents.

J. Lee, S. Schoenholz, J. Pennington, B. Adlam, L. Xiao, R. Novak, and J. Sohl-Dickstein. Finite versus infinite neural networks: an empirical study. *Advances in Neural Information Processing Systems*, 2020.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

G. Yang. Tensor programs iii: Neural matrix laws. *arXiv preprint arXiv:2009.10685*, 2020.

N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, pages 337–404, 1950.

C. Micchelli, Y. Xu, and H. Zhang. Universal kernels. *Journal of Machine Learning Research*, pages 2651–2667, 2006.

O. Kounchev. *Multivariate Polysplines: Applications to Numerical and Wavelet Analysis*. Elsevier Science, 2001.

J. Bradbury, R. Frostig, P. Hawkins, M. Johnson, C. Leary, D. Maclaurin, and S. Wanderman-Milne. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015.

R. Novak, L. Xiao, J. Lee, Y. Bahri, G. Yang, J. Hron, D. A Abolafia, J. Pennington, and J. Sohl-Dickstein. Bayesian deep convolutional networks with many channels are Gaussian processes. In *International Conference on Learning Representations*, 2019.

C. Wang, G. Zhang, and R. Grosse. Picking winning tickets before training by preserving gradient flow. In *International Conference on Learning Representations*, 2020.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019.

S De and SL Smith. Batch normalization biases residual blocks towards the identity function in deep networks. *Advances in Neural Information Processing Systems*, 2020.

H. Zhang, Y. N Dauphin, and T. Ma. Fixup initialization: Residual learning without normalization. In *International Conference on Learning Representations*, 2019.

T.M. MacRobert. *Spherical Harmonics: An Elementary Treatise on Harmonic Functions with Applications*. Pergamon Press, 1967.