Scaling Properties of Deep Residual Networks

Alain–Sam Cohen [1] **Rama Cont** [2] **Alain Rossier** [2 1] **Renyuan Xu** [2]

Abstract

Residual networks (ResNets) have displayed impressive results in pattern recognition and, recently, have garnered considerable theoretical interest due to a perceived link with neural ordinary differential equations (neural ODEs). This link relies on the convergence of network weights to a smooth function as the number of layers increases. We investigate the properties of weights trained by stochastic gradient descent and their scaling with network depth through detailed numerical experiments. We observe the existence of scaling regimes markedly different from those assumed in neural ODE literature. Depending on certain features of the network architecture, such as the smoothness of the activation function, one may obtain an alternative ODE limit, a stochastic differential equation or neither of these. These findings cast doubts on the validity of the neural ODE model as an adequate asymptotic description of deep ResNets and point to an alternative class of differential equations as a better description of the deep network limit.

1. Introduction

Residual networks, or ResNets, are multilayer neural network architectures in which a *skip connection* is introduced at every layer (He et al., 2016). This allows deep networks to be trained by circumventing vanishing and exploding gradients (Bengio et al., 1994). The increased depth in ResNets has lead to commensurate performance gains in applications ranging from speech recognition (Heymann et al., 2016; Zagoruyko & Komodakis, 2016) to computer vision (He et al., 2016; Huang et al., 2016).

A residual network with L layers may be represented as

$$h_{k+1}^{(L)} = h_k^{(L)} + \delta_k^{(L)} \sigma_d \left(A_k^{(L)} h_k^{(L)} + b_k^{(L)} \right), \qquad (1)$$

[1] InstaDeep [2]Mathematical Institute, University of Oxford. Correspondence to: Alain Rossier <rossier@maths.ox.ac.uk>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

where $h_k^{(L)}$ is the hidden state at layer $k = 0, \ldots, L$, $h_0^{(L)} = x \in \mathbb{R}^d$ the input, $h_L^{(L)} \in \mathbb{R}^d$ the output, $\sigma\colon \mathbb{R} \to \mathbb{R}$ is a non-linear activation function, $\sigma_d(x) = (\sigma(x_1), \ldots, \sigma(x_d))^\top$ its component-wise extension to $x \in \mathbb{R}^d$, and $A_k^{(L)}$, $b_k^{(L)}$, and $\delta_k^{(L)}$ are trainable network weights for $k = 0, \ldots, L-1$.

1.1. Connection to previous work

ResNets have been the focus of several theoretical studies due to a perceived link with a class of differential equations. The idea, put forth in (Haber & Ruthotto, 2018; Chen et al., 2018), is to view (1) as a discretization of a system of ordinary differential equations

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \sigma_d \left(\overline{A}_t H_t + \bar{b}_t \right), \qquad (2)$$

where $\overline{A}\colon [0,1] \to \mathbb{R}^{d\times d}$ and $\bar{b}\colon [0,1] \to \mathbb{R}^d$ are appropriate smooth functions and $H(0) = x$. This may be justified (Thorpe & van Gennip, 2018) by assuming that

$$\delta^{(L)} \sim 1/L, \quad A_k^{(L)} \to \overline{A}_t, \quad b_k^{(L)} \to \bar{b}_t \qquad (3)$$

as L increases and $k/L \to t$. Such models, named neural ordinary differential equations or neural ODEs (Chen et al., 2018; Dupont et al., 2019), have motivated the use of optimal control methods to train ResNets (E et al., 2019a).

However, the precise link between deep ResNets and the neural ODE model (2) is unclear: in practice, the weights $A^{(L)}$ and $b^{(L)}$ result from training, and the validity of the scaling assumptions (3) for trained weights is far from obvious and has not been verified. As a matter of fact, there is empirical evidence showing that using a scaling factor $\delta^{(L)} \sim 1/L$ can deteriorate the network accuracy (Bachlechner et al., 2020). Also, there is no guarantee that weights obtained through training have a non-zero limit which depends smoothly on the layer, as (3) would require. In fact, we present numerical experiments which point to the contrary for many ResNet architectures used in practice. These observations motivate an in-depth examination of the actual scaling behavior of weights with network depth in ResNets and of its impact on the asymptotic behavior of those networks.

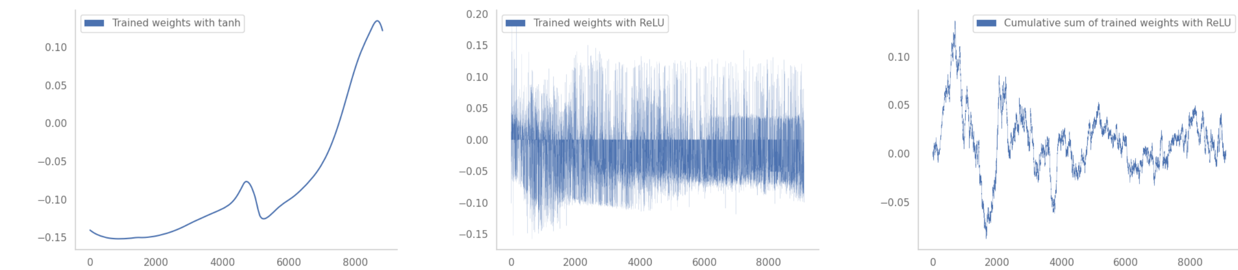

Figure 1: Trained weights as a function of k for $k = 0, \dots, L$ and $L = 9100$. **Left**: rescaled weights $L^{\beta} A^{(L)}_{k,(0,0)}$ for a $\tanh$ network with $\beta = 0.2$. **Center**: weights $A^{(L)}_{k,(0,0)}$ for a ReLU network. **Right**: cumulative sum $\sum_{j=0}^{k-1} A^{(L)}_{j,(0,0)}$ for a ReLU network.

1.2. Our contributions

We systematically investigate the scaling behavior of trained ResNet weights as the number of layers increases and examine the consequences of this behavior for the asymptotic properties of deep ResNets. Our code is publicly available at `https://github.com/instadeepai/scaling-resnets`.

Our main contributions are twofold. Using the methodology described in Section 2, we design detailed numerical experiments to study the scaling of trained network weights across a range of ResNet architectures and datasets, showing the existence of at least three different scaling regimes, none of which correspond to (3). In Section 4, we show that in two of these scaling regimes, the properties of deep ResNets may be described in terms of a class of ordinary or stochastic differential equations, albeit different from the neural ODEs studied in (Chen et al., 2018; Haber & Ruthotto, 2018; Lu et al., 2020). Those novel findings on the relation between ResNets and differential equations complement previous work (Thorpe & van Gennip, 2018; E et al., 2019b; Frei et al., 2019; Ott et al., 2021). In particular, our findings question the validity of the neural ODE (2) as a description of deep ResNets with trained weights.

1.3. Notations

$\|y\|$ denotes the Euclidean norm of a vector y. For a matrix x, $x^\top$ denotes its transpose, $\mathrm{Diag}(x)$ its diagonal vector, $\mathrm{Tr}(x)$ its trace and $\|x\|_F = \sqrt{\mathrm{Tr}(x^\top x)}$ its Frobenius norm. $\lfloor u \rfloor$ denotes the integer part of a positive number u. $\mathcal{N}(m, v)$ denotes the Gaussian distribution with mean m and (co)variance v, $\otimes$ denotes the tensor product, and $\mathbb{R}^{d,\otimes n} = \mathbb{R}^d \times \cdots \times \mathbb{R}^d$ (n times). $\mathrm{vec}\colon \mathbb{R}^{d_1 \times \cdots \times d_n} \to \mathbb{R}^{d_1 \cdots d_n}$ denotes the vectorisation operator, and $\mathbb{1}_S$ the indicator function of a set S. $\mathcal{C}^0$ is the space of continuous functions, and for $\nu \geq 0$, $\mathcal{C}^\nu$ is the space of ν-Hölder continuous functions.

2. Methodology

We identify the possible scaling regimes for the network weights, introduce the quantities needed to characterize the deep network limit, and describe the step-by-step procedure we use to analyze our numerical experiments.

2.1. Scaling hypotheses

As described in Section 1, the neural ODE limit assumes

$$\delta^{(L)} \sim \frac{1}{L} \quad \text{and} \quad A^{(L)}_{\lfloor Lt \rfloor} \overset{L\to\infty}{\longrightarrow} \overline{A}_t, \quad b^{(L)}_{\lfloor Lt \rfloor} \overset{L\to\infty}{\longrightarrow} \bar{b}_t, \quad (4)$$

for $t \in [0, 1]$ where $\overline{A}\colon [0, 1] \to \mathbb{R}^{d\times d}$ and $\bar{b}\colon [0, 1] \to \mathbb{R}^{d\times d}$ are smooth functions (Thorpe & van Gennip, 2018). Our numerical experiments, detailed in Section 3, show that the weights generally shrink as L increases (see for example Figures 2 and 4), so one cannot expect the above assumption to hold, and weights need to be renormalized in order to converge to a non-zero limit. We consider here the following more general situation which includes (4) but allows for shrinking weights:

Hypothesis 1. *There exist $\overline{A} \in \mathcal{C}^0\left([0, 1], \mathbb{R}^{d\times d}\right)$ and $\beta \in [0, 1]$ such that*

$$\forall s \in [0, 1], \qquad \overline{A}_s = \lim_{L\to\infty} L^\beta A^{(L)}_{\lfloor Ls \rfloor}. \qquad (5)$$

Properly renormalized weights may indeed converge to a continuous function of the layer in some cases, as shown in Figure 1 (left) which displays an example of layer dependence of trained weights for a ResNet (1) with fully connected layers and $\tanh$ activation function, without explicit regularization (see Section 3.1).

However it is not always the case that network weights converge to a smooth function of the layer, even after rescaling. For example, network weights $A^{(L)}_k$ are usually initialized to random, independent and identically distributed (i.i.d.) values, whose scaling limit would then correspond to a *white noise*, which cannot be represented as a function of the layer. Such scaling behaviour also occurs for trained weights, as shown in Figure 1 (center). In this case, the *cumulative sum* $\sum_{j=0}^{k-1} A_j^{(L)}$ of the weights behaves like a random walk, which does have a well-defined scaling limit $W \in \mathcal{C}^0\left([0,1], \mathbb{R}^{d\times d}\right)$. Figure 1 (right) shows that, for a ReLU ResNet with fully-connected layers, this cumulative sum of trained weights converges to an *irregular*, that is, non-smooth function of the layer.

This observation motivates the consideration of an alternative hypothesis where the weights $A_k^{(L)}$ are represented as the *increments* of a continuous function W^A.

Combining such terms with the ones considered in Hypothesis 1, we consider the following, more general, setting:

Hypothesis 2. *There exist $\beta \in [0,1)$, $\overline{A} \in \mathcal{C}^0\left([0,1], \mathbb{R}^{d\times d}\right)$, and $W^A \in \mathcal{C}^0([0,1], \mathbb{R}^{d\times d})$ non-zero such that $W_0^A = 0$ and*

$$A_k^{(L)} = L^{-\beta}\overline{A}_{k/L} + W^A_{(k+1)/L} - W^A_{k/L}. \tag{6}$$

The above decomposition is unique. Indeed, for $s \in [0,1]$,

$$\begin{aligned} L^{\beta-1}\sum_{k=0}^{\lfloor Ls\rfloor-1} A_k^{(L)} &= L^{-1}\sum_{k=0}^{\lfloor Ls\rfloor-1} \overline{A}_{k/L} + L^{\beta-1}W^A_{\lfloor Ls\rfloor/L} \\ &\to \int_0^s \overline{A}_r \mathrm{d}r, \quad \text{as } L\to\infty. \end{aligned} \tag{7}$$

The integral of $\overline{A}$ is thus uniquely determined by the weights $A_k^{(L)}$ when L is large, so $\overline{A}$ can be obtained by discretization and W^A by fitting the residual error in (7). In addition, Hypotheses 1 and 2 are mutually exclusive since Hypothesis 2 requires W^A to be non-zero.

Remark 2.1 (IID initialization of weights)**.** *In the special case of independent Gaussian weights*

$$A_{k,mn}^{(L)} \overset{i.i.d}{\sim} \mathcal{N}\left(0, L^{-1}d^{-2}\right) \quad \text{and} \quad b_{k,n}^{(L)} \overset{i.i.d}{\sim} \mathcal{N}\left(0, L^{-1}d^{-1}\right)$$

where $A_{k,mn}^{(L)}$ is the (m,n)-th entry of $A_k^{(L)} \in \mathbb{R}^{d\times d}$ and $b_{k,n}^{(L)}$ is the n-th entry of $b_k^{(L)} \in \mathbb{R}^d$, we can represent the weights $\{A^{(L)}, b^{(L)}\}$ as the increments of a matrix Brownian motion

$$A_k^{(L)} = d^{-1}\left(W^A_{(k+1)/L} - W^A_{k/L}\right),$$

which is a special case of Hypothesis 2.

2.2. Smoothness of weights with respect to the layer

A question related to the existence of a scaling limit is the degree of smoothness of the limits $\overline{A}$ or W^A, if they exist. To quantify the smoothness of the function mapping the layer number to the corresponding network weight, we define in Table 1 several quantities which may be viewed as discrete versions of various (semi-)norms used to measure the smoothness of functions.

Table 1: Quantities associated to a tensor $A^{(L)} \in \mathbb{R}^{L\times d\times d}$.

Quantity	**Definition**
Maximum norm	$\max_k \left\Vert A_k^{(L)}\right\Vert_F$
β-scaled norm of increments	$L^\beta \max_k \left\Vert A_{k+1}^{(L)} - A_k^{(L)}\right\Vert_F$
Cumulative sum norm	$\left\Vert \sum_k A_k^{(L)}\right\Vert_F$
Root sum of squares	$\left(\sum_k \left\Vert A_k^{(L)}\right\Vert_F^2\right)^{1/2}$

The first two norms relate to Hypothesis 1: if $A^{(L)}$ satisfy (5), then the maximum norm scales like $L^{-\beta}$ and the β-scaled norm of increments scales like $L^{-\nu}$ if the limit function $\overline{A}$ is ν-Hölder-continuous.
The last two norms relate to Hypothesis 2: if $A^{(L)}$ satisfy (6), then the cumulative sum norm scales like $L^{-\beta}$. Furthermore, the root sum of squares gives us the regularity of W^A. Indeed, define the *quadratic variation tensor* of W^A by

$$\left[W^A\right]_s = \lim_{L\to\infty}\sum_{k=0}^{\lfloor Ls\rfloor-1}\left(W^A_{\frac{k+1}{L}} - W^A_{\frac{k}{L}}\right)\otimes\left(W^A_{\frac{k+1}{L}} - W^A_{\frac{k}{L}}\right)^\top.$$

Then, using (6) and Cauchy-Schwarz, we estimate

$$\left\vert\kern-0.25ex\left\vert\kern-0.25ex\left\vert \left[W^A\right]_s \right\vert\kern-0.25ex\right\vert\kern-0.25ex\right\vert \leq 2\cdot\lim_{L\to\infty}\sum_{k=0}^{\lfloor Ls\rfloor-1}\left\Vert A_k^{(L)}\right\Vert_F^2 + L^{1-2\beta}\left\Vert\overline{A}\right\Vert_{L^2}^2 \tag{8}$$

where $\vert\kern-0.25ex\vert\kern-0.25ex\vert\cdot\vert\kern-0.25ex\vert\kern-0.25ex\vert$ is the Hilbert-Schmidt norm. As $\overline{A}$ is continuous on a compact domain, its L^2 norm is finite. Hence, if $\beta \geq 1/2$, the fact that the root sum of squares of $A^{(L)}$ is upper bounded as $L\to\infty$ implies that the quadratic variation of W^A is finite.

2.3. Procedure for numerical experiments

Note that Hypotheses 1 and 2 are mutually exclusive since Hypothesis 2 requires W^A to be non-zero. In order to examine whether one of these hypotheses, or neither, holds for the trained weights $A^{(L)}$ and $b^{(L)}$, we proceed as follows.

Step 1: We perform a logarithmic regression of the maximum norm of $\delta^{(L)}$ with respect to L to deduce the scaling $\delta^{(L)} \sim L^{-\alpha}$.

Step 2: To obtain the exponent $\beta \in [0,1)$, we perform a logarithmic regression of the cumulative sum norm of $A^{(L)}$ with respect to L. Indeed, (7) for $s = 1$ indicates that the cumulative sum norm explodes with a slope of $1-\beta$.

Step 3: After identifying the correct exponent β for the weights, we compute the β-scaled norm of increments of $A^{(L)}$ to check Hypothesis 1 and measure the smoothness of the trained weights. On one hand, if the β-scaled norm of increments of $A^{(L)}$ does not vanish as $L \to \infty$, it means that the rescaled weights cannot be represented as a continuous function of the layer, as in Hypothesis 1. On the other hand, if the β-scaled norm of increments of $A^{(L)}$ vanishes (say, as $L^{-\nu}$) when L increases, it supports Hypothesis 1 with a Hölder-continuous limit function $\overline{A} \in \mathcal{C}^{\nu}([0,1], \mathbb{R}^{d\times d})$.

Step 4: To discriminate between Hypothesis 1 and Hypothesis 2, we decompose the cumulative sum $\sum_{j=0}^{k-1} A_j^{(L)}$ of the trained weights into a *trend* component $\overline{A}$ and a *noise* component W^A, as shown in (7). The presence of non-negligible noise term W^A favors Hypothesis 2.

Step 5: Finally, we estimate the regularity of the term W^A under Hypothesis 2. If $\beta \geq 1/2$ and the root sum of squares of $A^{(L)}$ is finite, we deduce by (8) that W^A is of finite quadratic variation. It happens for example when W^A has a *diffusive* behavior, as in the example of i.i.d. random weights.

The same procedure is used for $b^{(L)}$. Note that the scaling exponent β may be different for $A^{(L)}$ and $b^{(L)}$.

Remark 2.2. *Note that $\sigma = \mathrm{ReLU}$ is homogeneous of degree 1, so we can write*

$$\delta \cdot \sigma_d\left(Ah + b\right) = \mathrm{sign}(\delta) \cdot \sigma_d\left(|\delta|\, Ah + |\delta|\, b\right).$$

Hence, when analyzing the scaling of trained weights in the case of a ReLU activation with fully-connected layers, we look at the quantities $\left|\delta^{(L)}\right| A^{(L)}$ and $\left|\delta^{(L)}\right| b^{(L)}$, as they represent the total scaling of the residual connection.

3. Numerical Experiments

We investigate the scaling properties and asymptotic behavior of trained weights for residual networks as the number of layers increases. We focus on two types of architectures: fully-connected and convolutional networks.

3.1. Fully-connected layers

Architecture. We consider a regression problem where the network layers are fully-connected. We consider the network architecture (1) for two different setups:

(i) $\sigma = \tanh$, $\delta_k^{(L)} = \delta^{(L)} \in \mathbb{R}_+$ trainable,

(ii) $\sigma = \mathrm{ReLU}$, $\delta_k^{(L)} \in \mathbb{R}$ trainable.

We choose to present these two cases for the following reasons. First, both $\tanh$ and ReLU are widely used in practice. Further, having $\delta^{(L)}$ scalar makes the derivation of the limiting behavior simpler. Also, since $\tanh$ is an odd function, the sign of $\delta^{(L)}$ can be absorbed into the activation. Therefore, we can assume that $\delta^{(L)}$ is non-negative for $\tanh$. Regarding ReLU, having a shared $\delta^{(L)}$ would hinder the expressiveness of the network. Indeed, if for instance $\delta^{(L)} > 0$, we would get $h_{k+1}^{(L)} \geq h_k^{(L)}$ element-wise since ReLU is non-negative. This would imply that $h_L^{(L)} \geq x$, which is not desirable. The same argument applies to the case $\delta^{(L)} < 0$. Thus, we let $\delta_k^{(L)} \in \mathbb{R}$ depend on the layer number for ReLU networks.

Datasets and training. We consider two datasets. The first one is synthetic: fix $d = 10$ and generate N i.i.d samples x_i coming from the $d-$dimensional uniform distribution in $[-1,1]^d$. Let $K = 100$ and simulate the following dynamical system:

$$\begin{cases} z_0^{x_i} &= x_i \\ z_k^{x_i} &= z_{k-1}^{x_i} + K^{-1/2} \tanh_d\left(g_d\left(z_{k-1}^{x_i}, k, K\right)\right), \end{cases}$$

where $g_d(z,k,K) := \sin(5k\pi/K)z + \cos(5k\pi/K)\mathbb{1}_d$. The targets y_i are defined as $y_i = z_K^{x_i} / \|z_K^{x_i}\|$. The motivation behind this low-dimensional dataset is to be able to train very deep residual networks on a problem where the optimal input-output map lies inside the class of functions represented by (1).

The second dataset is a low-dimensional embedding of the MNIST handwritten digits dataset (LeCun et al., 1998). Let $(\widetilde{x}, c) \in \mathbb{R}^{28\times 28} \times \{0,\dots,9\}$ be an input image and its corresponding class. We transform $\widetilde{x}$ into a lower dimensional embedding $x \in \mathbb{R}^d$ using an untrained convolutional projection, where $d = 25$. More precisely, we stack two convolutional layers initialized randomly, we apply them to the input and we flatten the downsized image into a $d-$dimensional vector. Doing so reduces the dimensionality of the problem while allowing very deep networks to reach at least 99% training accuracy. The target $y \in \mathbb{R}^d$ is the one-hot encoding of the corresponding class.

The weights are updated by stochastic gradient descent (SGD) on the unregularized mean-squared loss using batches of size B and a constant learning rate η. We perform SGD updates until the loss falls below ϵ, or when the maximum number of updates $T_{\max}$ is reached. We repeat the experiments for several depths L varying from $L_{\min}$ to $L_{\max}$. All the hyperparameters are given in Appendix A.

Results. For the case of a $\tanh$ activation (i), we observe in Figure 2 that for both datasets, $\delta^{(L)} \sim L^{-0.7}$ clearly decreases as L increases, and the cumulative sum norm of $A^{(L)}$ slightly increases when L increases. We deduce that $\beta = 0.2$ for the MNIST dataset and $\beta = 0.3$ for the synthetic dataset.

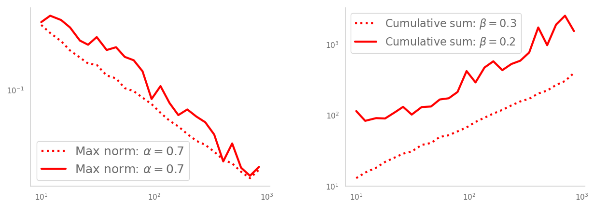

Figure 2: Scaling for $\tanh$ activation and $\delta^{(L)} \in \mathbb{R}$. Left: Maximum norm of $\delta^{(L)}$ with respect to L. Right: Cumulative sum norm of $A^{(L)}$ with respect to L. The dashed lines are for the synthetic data and the solid lines are for MNIST. The plots are in log-log scale.

Next, we verify which of Hypothesis 1 or Hypothesis 2 holds for $A^{(L)}$. We observe in Figure 3 (left) that the β-scaled norm of increments of $A^{(L)}$ decreases like $L^{-1/2}$, suggesting that Hypothesis 1 holds, with $\overline{A}$ being $1/2-$Hölder continuous. This is confirmed in Figure 3 (right), as the trend part $\overline{A}$ is visibly continuous and even of class $\mathcal{C}^1$. The noise part W^A is negligible. This observation is even more striking given that the weights are trained **without explicit regularization**.

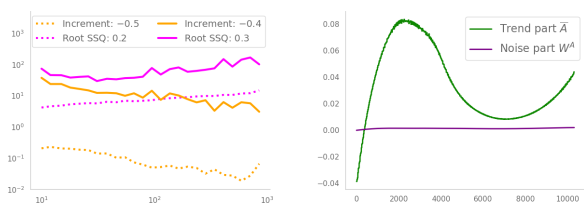

Figure 3: Hypothesis verification for $\tanh$ activation and $\delta^{(L)} \in \mathbb{R}$. Left: in pink we plot in log-log scale the root sum of squares of $A^{(L)}$, and in orange the β-scaled norm of increments of $A^{(L)}$. The dashed lines are for the synthetic data and the solid lines are for MNIST. Right: Decomposition of the trained weights $A^{(L)}_{k,(9,7)}$ with the trend part $\overline{A}$ and the noise part W^A for $L = 10321$, as defined in (6), for the synthetic dataset.

Regarding the case of a ReLU activation function (ii), we observe in Figure 4 (left) that the cumulative sum norm of the residual connection $\left|\delta^{(L)}\right| A^{(L)}$ scales like $L^{0.2}$ for the synthetic dataset and like $L^{0.1}$ for the MNIST dataset, so $\beta = 0.8$, resp. 0.9 in this case. We see in Figure 4 (right) that keeping the sign of $\delta^{(L)}_k$ is important, as the sign oscillates considerably throughout the network depth $k = 0, \ldots, L-1$.

We verify now which of Hypothesis 1 or Hypothesis 2 holds for $\left|\delta^{(L)}\right| A^{(L)}$. Figure 5 (left) shows that the β-scaled norm of increments scales like $L^{0.2}$ and $L^{0.4}$ as the depth increases. This suggests that there exists a noise part W^A. Following (8), the fact that the root sum of squares of $\left|\delta^{(L)}\right| A^{(L)}$ is upper bounded as $L \to \infty$ implies that W^A has finite quadratic variation. These claims are also

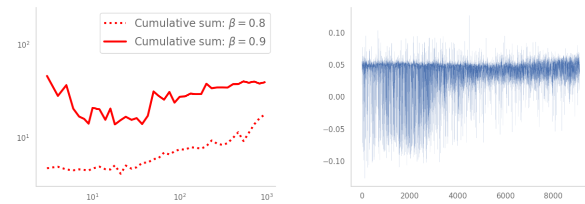

Figure 4: Scaling for ReLU activation and $\delta^{(L)}_k \in \mathbb{R}$. Left: Cumulative sum norm of $|\delta^{(L)}|A^{(L)}$ with respect to L, in log-log scale. Right: trained values of $\delta^{(L)}_k$ as a function of k, for $L = 9100$ and for the synthetic dataset.

supported by Figure 5 (right): there is a non-zero trend part $\overline{A}$, and a non-negligible noise part W^A.

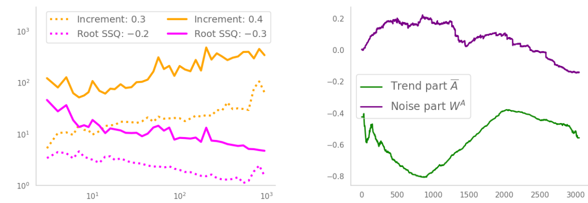

Figure 5: Hypothesis verification for ReLU activation and $\delta^{(L)}_k \in \mathbb{R}$. Left: in pink we plot in log-log scale the root sum of squares of $|\delta^{(L)}|A^{(L)}$, and in orange the β-scaled norm of increments of $|\delta^{(L)}|A^{(L)}$. The dashed lines are for the synthetic data and the solid lines for MNIST. Right: Decomposition of the trained weights $|\delta^{(L)}|\, A^{(L)}_{k,(7,7)}$ with the trend part $\overline{A}$ and the noise part W^A for $L = 10321$, as defined in (6), for the synthetic dataset.

Given the scaling behavior of the trained weights, we conclude that Hypothesis 1 seems to be a plausible description for the $\tanh$ case (i), but Hypothesis 2 provides a better description for the ReLU case (ii).

The same conclusions hold for $b^{(L)}$ as well, see Appendix B.

Role of the noise term W^A. A legitimate question to ask at this point is whether the noise part W^A plays a significant role in the accuracy of the network. To test this, we create a residual network with denoised weights $\widetilde{A}^{(L)}_k := L^{-\beta}\overline{A}_{k/L}$, compute its training error and we compare it to the original training error. We observe in Figure 6 (left) that for $\tanh$, the noise part W^A is negligible and does not influence the loss. However, for ReLU, the loss with denoised weights is one order of magnitude above the original training loss, meaning that the noise part W^A plays a significant role in the accuracy of the trained network.

3.2. Convolutional layers

We now consider the original ResNet with convolutional layers introduced in (He et al., 2016). This architecture is close to the state-of-the-art methods used for image recog-

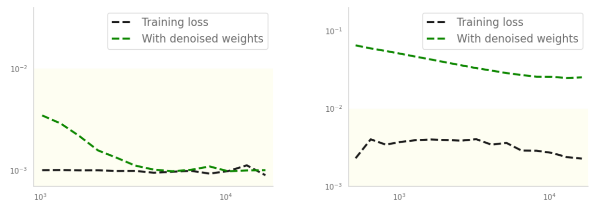

Figure 6: Loss value, as a function of L, in black for the trained weights $A_k^{(L)}$ and in green for the denoised weights $\widetilde{A}_k^{(L)} = L^{-\beta}\overline{A}_{k/L}$. Left: $\tanh$ activation and $\delta^{(L)} \in \mathbb{R}$. Right: ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$. Note that these curves are for the synthetic dataset and that we plot them in log-log scale. Also, we show in off-white the loss value range in which we consider our networks to have converged.

nition. In particular, we do not include batch normalization (Ioffe & Szegedy, 2015) since it only slightly improves the performance of the network while making the analysis significantly more complicated.

Setup. The precise architecture is detailed in Appendix D. Most importantly, our network still possesses the key skip connections from (1). Simply, the update rule for the hidden state reads

$$h_{k+1} = \sigma\left(h_k + \Delta_k * \sigma\left(A_k * h_k\right) + F_k * h_k\right) \qquad (9)$$

for $k = 0, \ldots, L-1$, where $\sigma = \mathrm{ReLU}$. Here, Δ_k, A_k, and F_k are kernels and $*$ denotes convolution. Note that Δ_k plays the same role as $\delta_k^{(L)}$ from (1). To lighten the notation, we omit the superscripts x (the input) and L (the number of layers).

We train our residual networks at depths ranging from $L_{\min} = 8$ to $L_{\max} = 121$ on the CIFAR-10 (Krizhevsky et al.) dataset with the unregularized cross-entropy loss. Here, the depth is the number of residual connections. We underline that a network with $L_{\max} = 121$ is already very deep. As a comparison, a standard ResNet-152 (He et al., 2016) has depth $L = 50$ in our framework.

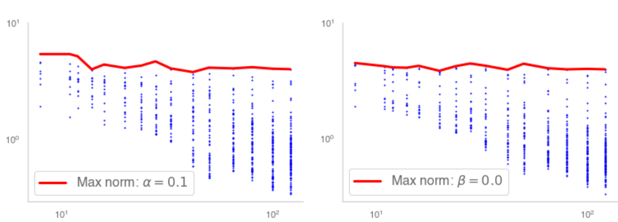

Figure 7: Scaling of $\Delta^{(L)}$ (left) and $A^{(L)}$ (right) against the network depth L for convolutional architectures on CIFAR-10. In blue, we plot the spectral norm of the kernels $\Delta_k^{(L)}$, resp. $A_k^{(L)}$, for $k = 0, \ldots, L-1$. The red line is the maximum of these values over k, namely the maximum norm, defined in Table 1. The plots are in log-log scale.

Results. As in Section 3.1, we investigate how the weights scale with network depth and whether Hypothesis 1 or Hypothesis 2 holds for a convolutional networks. To that end, we compute the spectral norms, of the linear operators defined by the convolutional kernels $\Delta_k^{(L)}$ and $A_k^{(L)}$ using the method described in (Sedghi et al., 2019). Figure 7 shows the maximum norm, and hence the scaling of $\Delta^{(L)}$ and $A^{(L)}$ against the network depth L. We observe that $\Delta^{(L)} \sim L^{-\alpha}$ and $A^{(L)} \sim L^{-\beta}$ with $\alpha = 0.1$ and $\beta = 0$.

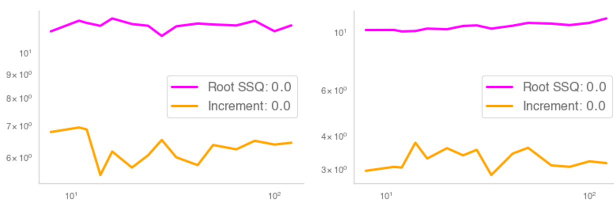

Figure 8: Testing of Hypotheses 1 and 2 for $\Delta^{(L)}$ (left) and $A^{(L)}$ (right). We plot in pink the root sum of squares and in orange the α-scaled norm of increments of $\Delta^{(L)}$ (left) and the β-scaled norm of increments of $A^{(L)}$ (right). Plots are in log-log scale. The root sum of squares and the scaled norm of increments are defined in Table 1. We obtain α and β from Figure 7.

We then use the values obtained for α and β to verify our hypotheses. Figure 8 shows that both the α-scaled norm of increments of $\Delta^{(L)}$ and the β-scaled norm of increments of $A^{(L)}$ seem to have lower bounds as the depth grows. This suggests that Hypothesis 1 does not hold for convolutional layers.

We also observe that the root sum of squares stays in the same order as the depth increases. Coupled with the fact that the maximum norms of $\Delta^{(L)}$ and $A^{(L)}$ are close to constant order as the depth increases, this suggests that the scaling limit is sparse with a finite number of weights being of constant order in L.

3.3. Summary: three scaling regimes

Our experiments show different scaling behaviors depending on the network architecture, especially the smoothness of the activation function. In Section 3.1, for fully-connected layers with $\tanh$ activation and a common pre-factor $\delta^{(L)}$ across layers, we observe a behavior consistent with Hypothesis 1 for both the synthetic dataset and MNIST. In contrast, a ResNet with fully-connected layers with ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$ shows behavior compatible with Hypothesis 2 both for the synthetic dataset and MNIST.

In the case of convolutional architectures trained on CIFAR-10 (Section 3.2) we observe that the maximum norm of the trained weights does not decrease with the network depth and the trained weights display a sparse structure, indicating a third scaling regime corresponding to a sparse structure for both $\Delta^{(L)}$ and $A^{(L)}$. These results are consistent with previous evidence on the existence of sparse CNN representations for image recognition (Mallat, 2016). We stress that the setup for our CIFAR-10 experiments has been chosen to approach state-of-the-art test performance with our generic architecture, as shown in Appendix C.

Note that the reason we consider networks with many layers (up to $L = 10321$) is to investigate the behavior of coefficients as depth varies, not because of any claim that very deep networks are more robust or generalize better than shallower ones. In fact, most of the networks exhibit good accuracy for depths $L \geq 15$.

4. Deep Network Limit

In this section, we analyze the scaling limit of the residual network (1) under Hypotheses 1 and 2.

4.1. Setup and assumptions

We consider $\delta^{(L)} = L^{-\alpha}$ for some $\alpha \geq 0$ and

$$\begin{aligned} h_0^{(L)} &= x, \\ h_{k+1}^{(L)} &= h_k^{(L)} + L^{-\alpha}\,\sigma_d\left(A_k^{(L)} h_k^{(L)} + b_k^{(L)}\right), \end{aligned} \tag{10}$$

with

$$\begin{aligned} A_k^{(L)} &= L^{-\beta}\overline{A}_{k/L} + W^A_{(k+1)/L} - W^A_{k/L}, \\ b_k^{(L)} &= L^{-\beta}\overline{b}_{k/L} + W^b_{(k+1)/L} - W^b_{k/L}, \end{aligned}$$

where $(W_t^A)_{t\in[0,1]}$ and $(W_t^b)_{t\in[0,1]}$ are Itô processes (Revuz & Yor, 2013) with regularity conditions specified in Appendix E.1.

Remark 4.1. *Hypothesis 1 corresponds to the case $W^A \equiv 0$ and $W^b \equiv 0$. Hypothesis 2 corresponds to the case where W^A and W^b are non-zero.*

We use the following notation for the quadratic variation of W^A and W^b:

$$\left[W^A\right]_t = \int_0^t \Sigma_u^A \mathrm{d}u, \quad \left[W^b\right]_t = \int_0^t \Sigma_u^b \mathrm{d}u, \tag{11}$$

where Σ^A and Σ^b are bounded processes with values respectively in $\mathbb{R}^{d,\otimes 4}$ and $\mathbb{R}^{d\times d}$. Let $Q_i : [0,1]\times\mathbb{R}^d \to \mathbb{R}$ be the (random) quadratic form defined by

$$Q_i(t,x) \coloneqq \sum_{j,k=1}^{d} x_j x_k \left(\Sigma_t^A\right)_{ijik} + \Sigma_{t,ii}^b. \tag{12}$$

and $Q(t,x) = (Q_1(t,x),\ldots,Q_d(t,x))$. Our analysis focuses on smooth activation functions.

Assumption 4.2 (Activation function)**.** *The activation function σ is in $\mathcal{C}^3(\mathbb{R},\mathbb{R})$ and satisfies $\sigma(0) = 0$, $\sigma'(0) = 1$. Moreover, σ has a bounded third derivative σ'''.*

Most smooth activation functions, including $\tanh$, satisfy this condition. Also, the boundedness of the third derivative σ''' could be further relaxed to some exponential growth condition, see (Peluchetti & Favaro, 2020).

Finally, we assume that the hidden state dynamics $(h_k^{(L)}, k = 1,\ldots,L)$ given by (1) is uniformly integrable. (For a precise statement see Assumption E.3 in Appendix.) This is a reasonable assumption since both the inputs and the outputs of the network are uniformly bounded.

4.2. Informal derivation of the deep network limit

We first provide an informal analysis on the derivation of the deep network limit. Denote $t_k = k/L$ and define for $s \in [t_k, t_{k+1}]$:

$$\begin{aligned} M_s^{(L),k} = &\left(W_s^A - W_{t_k}^A\right) h_k^{(L)} + W_s^b - W_{t_k}^b \\ &+ L^{1-\beta}\overline{A}_{t_k} h_k^{(L)}(s - t_k) + L^{1-\beta}\overline{b}_{t_k}(s-t_k). \end{aligned}$$

When $\beta = 0$, we need $\alpha = 1$ to obtain a non-trivial limit. In this case, the noise terms in $M_s^{(L),k}$ are vanishing as L increases, and the limit is the Neural ODE. See Lemma 4.6 in (Thorpe & van Gennip, 2018).

When $\beta > 0$ we can apply the Itô formula (Itô, 1944) to $\sigma\big(M_s^{(L),k}\big)$ for $s \in [t_k, t_{k+1})$ to obtain the approximation

$$h_{k+1}^{(L)} - h_k^{(L)} = \delta^{(L)}\sigma\left(M_{t_{k+1}}^{(L),k}\right) = D_1 + D_2 + D_3 + o\left(1/L\right) \tag{13}$$

where

$$\begin{aligned} D_1 &= L^{-\alpha}\left(\left(W_{t_{k+1}}^A - W_{t_k}^A\right) h_k^{(L)} + \left(W_{t_{k+1}}^b - W_{t_k}^b\right)\right) \\ D_2 &= \frac{1}{2} L^{-\alpha}\sigma''(0)\, Q\left(t_k, h_k^{(L)}\right)(t_{k+1} - t_k) \\ D_3 &= L^{1-\beta-\alpha}\left(\overline{A}_{t_k} h_k^{(L)}(t_{k+1} - t_k) + \overline{b}_{t_k}(t_{k+1} - t_k)\right). \end{aligned}$$

When $\alpha = 0$, we see from D_1 that (13) admits a diffusive limit, that is with non-vanishing noise terms W^A and W^b. In this case, D_2 stays bounded when L increases, and D_3 does not explode only when $\beta \geq 1$. The case $\alpha = 0, \beta \geq 1$ corresponds to a stochastic differential equation (SDE) limit.

When $\alpha > 0$, D_1 and D_2 vanish in the limit $L \to \infty$, and we need $\beta = 1 - \alpha$ to obtain a non-trivial ODE limit.

We provide precise mathematical statements of these results in the next section.

4.3. Statement of the results

The following results describe the different scaling limits of the hidden state dynamics $(h_k^{(L)}, k = 1,\ldots,L)$ for various values of scaling exponents α and β.

First, we show that Hypothesis 1 with a smooth activation function leads to convergence in sup norm to an ODE limit that is different from the neural ODE behavior described in (Chen et al., 2018; Thorpe & van Gennip, 2018; Haber & Ruthotto, 2018).

Theorem 4.3 (Asymptotic behavior under Hypothesis 1)**.** *If the activation function satisfies Assumption 4.2,* $0 < \alpha < 1$ *and* $\alpha + \beta = 1$*, then the hidden state converges to the solution to the ODE*

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \overline{A}_t H_t + \overline{b}_t, \qquad H_0 = x, \tag{14}$$

in the sense that

$$\lim_{L\to\infty} \sup_{0\le t\le 1} \left|\left| H_t - h^{(L)}_{\lfloor tL \rfloor} \right|\right| = 0.$$

In particular, this implies the convergence of the hidden state process for any typical initialization, i.e almost-surely with respect to the initialization.

Note that in Theorem 4.3, the limit (14) defines a linear input-output map behaving like a linear network (Arora et al., 2019). This is different from the neural ODE (2), where the activation function σ appears in the limit. Interestingly, the limit is a controlled ODE where the control parameters are linear in the derivative of the state. Their expressivity is discussed in (Cuchiero et al., 2020).

We show that under Hypothesis 2, we may obtain either an SDE or an ODE limit. In the latter case, the limiting ODE is found to be different from the neural ODE (2).

Theorem 4.4 (Asymptotic behavior under Hypothesis 2)**.** *Let* σ *be an activation function satisfying Assumption 4.2.* $\alpha = 0$ and $\beta \geq 1$: diffusive limit. *Let* H *be the solution to the SDE*

$$\begin{aligned}\mathrm{d}H_t = {} & H_t\,\mathrm{d}W^A_t + \mathrm{d}W^b_t + \frac{1}{2}\sigma''(0)Q(t,H_t)\,\mathrm{d}t \\ & + \mathbb{1}_{\beta=1}(\overline{A}_t H_t + \overline{b}_t)\,\mathrm{d}t,\end{aligned} \tag{15}$$

with initial condition $H_0 = x$*. If there exists* $p_2 > 2$ *such that* $\mathbb{E}\left[\sup_{0\le t\le 1} \|H_t\|^{p_2}\right] < \infty$*, then the hidden state converges uniformly in* L^2 *to the solution* H *of (15):*

$$\lim_{L\to\infty} \mathbb{E}\left[\sup_{0\le t\le 1} \left\| h^{(L)}_{\lfloor tL \rfloor} - H_t \right\|^2\right] = 0.$$

$0 < \alpha < 1$, $\alpha + \beta = 1$: ODE limit. *The hidden state converges uniformly in* L^2 *to the solution of the ODE*

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \overline{A}_t H_t + \overline{b}_t, \tag{16}$$

with initial condition $H_0 = x$*:*

$$\lim_{L\to\infty} \mathbb{E}\left[\sup_{0\le t\le 1} \left\| h^{(L)}_{\lfloor tL \rfloor} - H_t \right\|^2\right] = 0.$$

Note that we prove uniform convergence in L^2, also known as strong convergence.

The detailed assumptions and a sketch of the proof for Theorems 4.3 and 4.4 are given in Appendix E. Further details and some extensions may be found in the companion paper (Cohen et al., 2021).

The idea of the proof is to view the ResNet as a *nonlinear* Euler discretization of the limit equation, and then bound the difference between the hidden state and a classical Euler discretization. Then, using an extension of the techniques in (Higham et al., 2002) to the case of equations driven by Itô processes, we show strong convergence in the following way. We first show that the drift term of (15) is locally Lipschitz (Appendix E.3). We then prove the strong convergence of the hidden state dynamics (10) by bounding the difference between the hidden state and an Euler scheme for the limiting equation. It is worth mentioning that the convergence results in (Higham et al., 2002) hold for a class of time-homogeneous diffusion processes whereas our result holds for general Itô processes. This distinction is important for training neural networks since the diffusion assumption involves the Markov property which cannot be expected to hold after training with backpropagation.

In addition, we also relax a technical condition from (Higham et al., 2002), which is difficult to verify in practice. See Remark E.5.

4.4. Remarks on the results

Interestingly, when the activation function σ is smooth, all limits in both Theorems 4.3 and 4.4 depend on the activation only through $\sigma'(0)$ (assumed to be 1 for simplicity) and $\sigma''(0)$. Hypotheses 1 and 2 lead to the same ODE limit when $0 < \alpha < 1$ and $\alpha + \beta = 1$. In contrast to the neural ODE (2), the characteristics of σ away from 0 are not relevant to this limit. In addition, our proof relies on the smoothness of σ at 0. If the activation function is not differentiable at 0, then a different limit should be expected.

The case $\overline{A} \equiv 0$, $\overline{b} \equiv 0$, $\alpha = 0$, and $\beta = 1$ in Theorem 4.4 is considered in (Peluchetti & Favaro, 2020), who prove weak convergence under the additional assumption that W^A and W^b are Brownian motions with constant drift. In our setup, W^A and W^b are allowed to be arbitrary Itô processes, whose increments, i.e. the network weights, are not necessarily independent nor identically distributed. This allows for a general distribution and dependence structure of weights across layers.

The concept of weak convergence used in (Peluchetti & Favaro, 2020) corresponds to convergence of quantities averaged across many random IID weight initializations. In practice, as the training is done only once, the strong convergence, shown in Theorems 4.3 and 4.4 is a more relevant notion for studying the asymptotic behavior of deep neural networks.

4.5. Link with numerical experiments

Let us now discuss how the analysis above sheds light on the numerical results in Section 3.1 and Section 3.2.

Figure 2 shows that $\beta \approx 0.3$ and $\alpha \approx 0.7$ for the synthetic dataset with fully-connected layers and $\tanh$ activation function, and Figure 3 suggests that Hypothesis 1 is more likely to hold. This corresponds to the assumptions of Theorem 4.3 with the ODE limit (14). This is also consistent with the estimated decomposition in Figure 3 (right) where the noise part tends to be negligible.

In the case of ReLU activation with fully-connected layers, we observe that $\beta + \alpha \approx 1$ from Figure 4 (left). Since ReLU is homogeneous of degree 1 (see Remark 2.2), $|\delta^{(L)}|$ can be moved inside the activation function, so without loss of generality we can assume $\alpha = 0$ and $\beta \approx 1$. If we replace the ReLU function by a smooth version σ^{ϵ}, then the limit is described by the stochastic differential equation (15). The ReLU case would then correspond to a limit of this equation as $\epsilon \to 0$. The existence of such a limit is, however, nontrivial.

From the experiments with convolutional architectures, we observe that the maximum norm (Figure 7), the scaled norm of the increments, and the root sum of squares (Figure 8) are upper bounded as the number of layers L increases. This indicates that the weights become sparse when L is large. In this case, there is no continuous ODE or SDE limit and Hypotheses 1 and 2 both fail. This emergence of sparse representations in convolutional networks is consistent with previous results on such networks (Mallat, 2016).

5. Conclusion

We study the scaling behavior of trained weights in deep residual networks. We provide evidence for the existence of at least three different scaling regimes that encompass differential equations and sparse scaling limits. We also theoretically characterize the ODE and SDE limits for the hidden state dynamics in deep fully-connected residual networks.

Our work contributes to a better understanding of the behavior of residual networks and the role of network depth. Our findings point to interesting questions regarding the asymptotic behavior of such networks in the case of non-smooth activation functions and more complex architectures.

References

Arora, S., Cohen, N., Golowich, N., and Hu, W. A Convergence Analysis of Gradient Descent for Deep Linear Neural Networks. In *7th International Conference on Learning Representations (ICLR)*, 2019.

Bachlechner, T., Majumder, B. P., Mao, H. H., Cottrell, G. W., and McAuley, J. ReZero is All You Need: Fast Convergence at Large Depth, 2020.

Bengio, Y., Simard, P., and Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. doi: 10.1109/72.279181.

Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems (NeurIPS) 31*, pp. 6571–6583, 2018.

Cohen, A.-S., Cont, R., Rossier, A., and Xu, R. Asymptotic analysis of deep residual networks. Working paper, 2021.

Cuchiero, C., Larsson, M., and Teichmann, J. Deep neural networks, generic universal interpolation and controlled ODEs. *SIAM Journal on Mathematics of Data Science*, 2 (3):901–919, 2020.

Dupont, E., Doucet, A., and Teh, Y. W. Augmented Neural ODEs. In *Advances in Neural Information Processing Systems (NeurIPS) 32*, 2019.

E, W., Han, J., and Li, Q. A mean-field optimal control formulation of deep learning. *Research in the Mathematical Sciences*, 6(1):1–41, 2019a.

E, W., Ma, C., Wang, Q., and Wu, L. Analysis of the Gradient Descent Algorithm for a Deep Neural Network Model with Skip-connections. 2019b.

Frei, S., Cao, Y., and Gu, Q. Algorithm-Dependent Generalization Bounds for Overparameterized Deep Residual Networks. In *Advances in Neural Information Processing Systems (NeurIPS) 32*, 2019.

Haber, E. and Ruthotto, L. Stable architectures for deep neural networks. *Inverse Problems*, 34(1), 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778. IEEE Computer Society, 2016.

Heymann, J., Drude, L., and Haeb-Umbach, R. Wide Residual BLSTM Network with Discriminative Speaker Adaptation for Robust Speech Recognition. In *Computer Speech and Language*, 2016.

Higham, D. J., Mao, X., and Stuart, A. M. Strong convergence of Euler-type methods for nonlinear stochastic differential equations. *SIAM Journal on Numerical Analysis*, 40(3):1041–1063, 2002.

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Deep Networks with Stochastic Depth. In *European Conference on Computer Vision*, 2016.

Ioffe, S. and Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 448–456, 2015.

Itô, K. Stochastic integral. *Proc. Imp. Acad.*, 20(8):519–524, 1944. doi: 10.3792/pia/1195572786.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization, 2014.

Krizhevsky, A., Nair, V., and Hinton, G. CIFAR-10.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, volume 86, pp. 2278–2324, 1998.

Loshchilov, I. and Hutter, F. SGDR: Stochastic Gradient Descent with Warm Restarts. In *5th International Conference on Learning Representations (ICLR)*, 2017.

Lu, Y., Ma, C., Lu, Y., Lu, J., and Ying, L. A Mean Field Analysis Of Deep ResNet And Beyond: Towards Provably Optimization Via Overparameterization From Depth. In *International Conference on Machine Learning*, pp. 6426–6436. PMLR, 2020.

Mallat, S. Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065): 20150203, 2016. doi: 10.1098/rsta.2015.0203.

Ott, K., Katiyar, P., Hennig, P., and Tiemann, M. ResNet After All: Neural ODEs and Their Numerical Solution. In *Intl Conference on Learning Representations (ICLR)*, 2021.

Peluchetti, S. and Favaro, S. Infinitely deep neural networks as diffusion processes. In *Intl Conference on Artificial Intelligence and Statistics*, pp. 1126–1136. PMLR, 2020.

Platen, E. and Bruti-Liberati, N. *Numerical solution of stochastic differential equations with jumps in finance*, volume 64. Springer Science & Business Media, 2010.

Revuz, D. and Yor, M. *Continuous martingales and Brownian motion*. Springer, 2013.

Sedghi, H., Gupta, V., and Long, P. M. The Singular Values of Convolutional Layers. In *Intl Conference on Learning Representations (ICLR)*, 2019.

Thorpe, M. and van Gennip, Y. Deep Limits of Residual Neural Networks, 2018.

Zagoruyko, S. and Komodakis, N. Wide Residual Networks, 2016.