A Mean-field Analysis of Deep ResNet and Beyond: Towards Provable Optimization Via Overparameterization From Depth

Yiping Lu [1] Chao Ma [2] Yulong Lu [3] Jianfeng Lu [3] Lexing Ying [4]

Abstract

Training deep neural networks with stochastic gradient descent (SGD) can often achieve zero training loss on real-world tasks although the optimization landscape is known to be highly non-convex. To understand the success of SGD for training deep neural networks, this work presents a mean-field analysis of deep residual networks, based on a line of works that interpret the continuum limit of the deep residual network as an ordinary differential equation when the network capacity tends to infinity. Specifically, we propose a **new continuum limit** of deep residual networks, which enjoys a good landscape in the sense that **every local minimizer is global**. This characterization enables us to derive the first global convergence result for multilayer neural networks in the mean-field regime. Furthermore, without assuming the convexity of the loss landscape, our proof relies on a zero-loss assumption at the global minimizer that can be achieved when the model shares a universal approximation property. Key to our result is the observation that a deep residual network resembles a shallow network ensemble (Veit et al., 2016), *i.e.* a two-layer network. We bound the difference between the shallow network and our ResNet model via the adjoint sensitivity method, which enables us to apply existing mean-field analyses of two-layer networks to deep networks. Furthermore, we propose several novel training schemes based on the new continuous model, including one training procedure that switches the order of the residual blocks and results in strong empirical performance on the benchmark datasets.

1. Introduction

Neural networks have become state-of-the-art models in numerous machine learning tasks and strong empirical performance is often achieved by deeper networks. One landmark example is the residual network (ResNet) (He et al., 2016a;b), which can be efficiently optimized even at extremely large depth such as 1000 layers. However, there exists a gap between this empirical success and the theoretical understanding: ResNets can be trained to almost zero loss with standard stochastic gradient descent(Zhang et al., 2016; Ishida et al., 2020), yet it is known that larger depth leads to increasingly non-convex landscape even the the presence of residual connections (Yun et al., 2019). While global convergence can be obtained in the so-called "lazy" regime e.g. (Jacot et al., 2018; Du et al., 2018), such kernel models cannot capture fully-trained neural networks (Suzuki, 2018; Chizat et al., 2019; Ghorbani et al., 2019).

In this work, we aim to demonstrate the provable optimization of ResNet beyond the restrictive "lazy" regime. To do so, we build upon recent works that connect ordinary differential equation (ODE) models to infinite-depth neural networks (E, 2017; Lu et al., 2017; Sonoda & Murata, 2017; Haber & Ruthotto, 2017; Chen et al., 2018; Dupont et al., 2019; Zhang et al., 2019c; Thorpe & van Gennip, 2018; Sonoda & Murata, 2019; Lu et al., 2019). Specifically, each residual block of a ResNet can be written as $x_{n+1} = x_n + \Delta t f(x_n, \theta_n)$, which can be seen as the Euler discretization of the ODE $\dot{x}_t = f(x, t)$. This turns training the neural network into solving an optimal control problem (Li et al., 2017; E et al., 2019a; Liu & Theodorou, 2019), under which backpropagation can be understood as simulating the adjoint equation (Chen et al., 2018; Li et al., 2017; Li & Hao, 2018; Zhang et al., 2019a; Li et al., 2020). However, this analogy does not directly provide guarantees of global convergence even in the continuum limit.

To address the problem of global convergence, we propose **a new limiting ODE model** of ResNets. Formally, we model deep ResNets via a mean-field ODE model

$$\dot{X}_\rho(x,t) = \int_\theta f(X_\rho(x,t),\theta)\rho(\theta,t)d\theta$$

This model considers every residual block $f(\cdot, \theta_i)$ as a parti-

[*] Equal contribution [1] Institute for Computational & Mathematical Engineering, Stanford University [2] PACM, Princeton University [3] Mathematics Department, Duke University [4] Department of Mathematics, Stanford University. Correspondence to: Yiping Lu <yplu@stanford.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020. Copyright 2020 by the author(s).

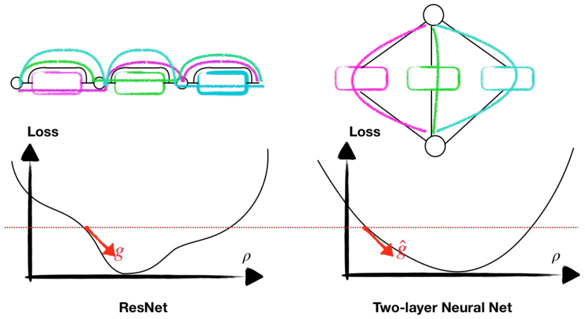

Figure 1. Illustration that ResNet behaves like shallow network ensemble, *i.e.* a two-layer overparameterized neural network. The high-level intuition is to show that the gradient of the two models are at the same scale when the loss are comparable.

cle and optimizes over the empirical distribution of particles $\rho(\theta, t)$, where θ denotes the weight of the residual block and t denotes the layer index of the residual block. Similar limiting objective function is proposed in (Hu et al., 2019; Jabir et al., 2019; Ma et al., 2019; E et al., 2019b). (Hu et al., 2019; Jabir et al., 2019) have introduce a further convex condition on the Hamiltonian function which is generally not true for the realistic setting. (Ma et al., 2019) is mainly discussing the statistical property of the objective which is out of the scope of the discussing of this paper. We consider properties of the loss landscape with respect to the distribution of weights, an approach similar to Bengio et al. (2006); Bach (2017). Inspired by (Veit et al., 2016) that a deep ResNet behaves like an ensemble of shallow models, we compare a deep ResNet with its counterpart two-layer network and show that the gradients of the two models are close to each other. This leads us to conclude that, although the loss landscape may not be convex, every local minimizer is a global one.

1.1. Contribution

Our contributions can be summarized as follows:

- We derive a new continuous depth limit of deep ResNets. In this new model, each residual block is regarded as a particle and the training dynamics is captured by the gradient flow on the distribution of the particles ρ.
- We analyze the loss landscape with respect to ρ and show that all local minima have zero loss, which indicates that every local optima is global. This property leads to the conclusion that a full support stationary point of the Wasserstein gradient flow is a global optimum. To the best of our knowledge, this is the **first global convergence result for multi-layer neural networks in the mean-field regime** without the convexity assumption on the loss landscape.
- We propose novel numerical schemes to approximate the mean-field limit of the deep ResNets and demonstrate that they achieves superior empirical results on real-world datasets.

1.2. Related Work

Mean-Field Limit and Global Convergence. Recent works have explored the global convergence of two-layer neural networks by studying suitable scaling limits of the stochastic gradient descent of two-layer neural network when the width is sent to infinity and the second layer scaled by one over the width of the neural network (Nitanda & Suzuki, 2017; Mei et al., 2018; Rotskoff & Vanden-Eijnden, 2018; Chizat & Bach, 2018; Sirignano & Spiliopoulos, 2019). Though global convergence can be obtained under certain conditions for two-layer networks, it is highly nontrivial to extend this framework to multi-layer neural networks: recent attempts (Araújo et al., 2019; Sirignano & Spiliopoulos, 2019; Nguyen, 2019; Fang et al., 2019) do not address realistic neural architectures directly or provide conditions for global convergence.

Parallel to the mean-field regime, Jacot et al. (2018); Du et al. (2018); Allen-Zhu et al. (2018); Zou et al. (2018); Oymak & Soltanolkotabi (2019) provided global convergence results for multi-layer networks in the so-called ”lazy” or kernel regime. However, this description of deep neural networks is rather limited: the scaling of initialization forces the distance traveled by each parameter to vanish asymptotically (Chizat et al., 2019), and thus training becomes equivalent to kernel regression with respect to *neural tangent kernel* (Arora et al., 2019; Jacot et al., 2018). On the other hand, it is well-known that properly trained neural networks can outperform kernel models in learning various target functions (Wei et al., 2019; Suzuki, 2018; Ghorbani et al., 2019; Ba et al., 2020; Allen-Zhu & Li, 2019). In contrast, the mean-field regime considered in this work does not reduce training into kernel regression; in other words, the mean-field setting allows neurons to travel further and learn adaptive features.

Landscape of ResNets. Li & Yuan (2017); Liu et al. (2019) provided convergence results of gradient descent on two-layer residual neural networks and showed that the global minimum is unique. In parallel, Shamir (2018); Kawaguchi & Bengio (2019) showed that when the network consists of one residual block the gradient descent solution is provably better than a linear classifier. However, recent work also pointed out that these positive results may not hold true for deep ResNets composed of multiple residual blocks. Regarding deeper models, Hardt & Ma (2016); Bartlett et al. (2019); Wu et al. (2019) proved the global convergence of the gradient descent for training deep *linear* ResNets. Yet it is known that even mild nonlinear activation functions can destroy these good landscape properties (Yun et al., 2018). In addition, (Bartlett et al., 2018) considered a ResNet model with compositions of close-to-identity functions, and provided convergence result regarding the Fréchet gradient. However, (Bartlett et al., 2018) also pointed out that such conclusion may no longer hold for a realistic ResNet model. Our paper fills this gap by introducing a new continuous model and providing conditions for the global convergence beyond the previously considered kernel regime (Du et al., 2018; Zhang et al., 2019b; Allen-Zhu et al., 2018; Zhang et al., 2019b).

1.3. Notations and Preliminaries

Notations. Let $\delta(\cdot)$ denote the Dirac mass and 1_Ω be the indicator function on Ω. We denote by $\mathcal{P}^2$ the set of probability measures endowed with the Wasserstein-2 distance (see below for definition). Let μ be the population distribution of the input data and the induced norm by $\|f\|_\mu = \sqrt{\mathbb{E}_{x\sim\mu}[f(x)^\top f(x)]}$.

Fréchet Derivative. We extend the notion of the gradient to infinite dimensional space. For a functional $f : X \to \mathbb{R}$ defined on a Banach space X, the Fréchet derivative is an element in the dual space $df \in X^*$ that satisfies

$$\lim_{\delta\in X,\delta\to 0} \frac{f(x+\delta) - f(x) - df(\delta)}{\|\delta\|} = 0, \quad \text{for all } x \in X.$$

In this paper, $\frac{\delta f}{\delta X}$ is used to denote the Fréchet derivative.

Wasserstein Space. The Wasserstein-2 distance between two probability measures $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$ is defined as

$$W_2(\mu,\nu) := \left(\inf_{\gamma\in\mathcal{T}(\mu,\nu)} \int |y-x|^2 d\gamma(x,y) \right)^{1/2}.$$

Here $\mathcal{T}(\mu,\nu)$ denotes the set of all couplings between μ and ν, i.e., all probability measures $\gamma \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d)$ with marginals μ on the first factor and ν on the second.

Bounded Lipschitz norm. We say that a sequence of measures $\mu_n \in (\mathbb{R}^d)$ *weakly* (or *narrowly*) converges to μ if, for all continuous and bounded function $\varphi : \mathbb{R}^d \to \mathbb{R}$ it holds $\int \varphi \underset{\cdot}{\mu}_{n} \to \int \varphi \underset{\cdot}{\mu}$. For sequences which are bounded in total variation norm, this is equivalent to the convergence in Bounded Lipschitz norm. The latter is defined, for $\mu \in (\mathbb{R}^d)$, as

$$\|\mu\|_{:} = \sup\left\{ \int \varphi \, \underset{\cdot}{\mu} \; ; \; \varphi : \mathbb{R}^d \to \mathbb{R}, \; (\varphi) \le 1, \; \|\varphi\|_\infty \le 1 \right\} \quad (1)$$

where (φ) is the smallest Lipschitz constant of φ and $\|\cdot\|_\infty$ the supremum norm.

2. Limiting Model

Following the observation that each residual block of a ResNet $u_{n+1} = u_n + \Delta t f(u_n, \theta_n)$ can be considered as one step of the forward Euler approximation of the ODE $u_t = f(u,t)$ (E, 2017; Lu et al., 2017; Sonoda & Murata, 2017; Haber & Ruthotto, 2017), a series of recent papers (Zhang et al., 2019c;a; Chen et al., 2018; Li et al., 2020; 2017; 2020) analyzed the deep neural networks in the continuous limit. Thorpe & van Gennip (2018) proved the Gamma-convergence of ResNets in the asymptotic limit. However, there are two points of that approach that require further investigation. First, Thorpe & van Gennip (2018) introduced a regularization term $n\sum_{i=1}^{n} \|\theta_i - \theta_{i-1}\|^2$, where n is the depth of the network. This regularization becomes stronger as the network gets deeper, which implies a more constrained space of functions that the network can represent.

Second, while the Gamma-convergence result is concerned with the convergence of the global minima of a sequence of energy functionals, it gives rather little information about the landscape of the limiting functional, which can be quite complicated for non-convex objective functions. Later work (Avelin & Nyström, 2019) proved that stochastic gradient descent of a deep ResNet with constant weight across layers converges to the gradient flow of loss using the ODE model. However, letting the weights of the ResNet be the same across all layers weakens the approximation power and makes optimization landscape more complicated. To address the reason behind the global convergence of the gradient flow, in this section, we propose a new continuous limiting model of the deep residual network.

2.1. A New Continuous Model

The goal is to minimize the l_2 loss function

$$E(\rho) = \mathbb{E}_{x\sim\mu}\Big[\frac{1}{2}\left(\langle w_1, X_\rho(x,1)\rangle - y(x)\right)^2\Big]. \quad (2)$$

over parameter distributions $\rho(\theta,t)$ for θ in a compact set Ω and $t \in [0,1]$. Here $X_\rho(x,t)$ is the solution of the ODE

$$\dot{X}_\rho(x,t) = \int_\theta f(X_\rho(x,t),\theta)\rho(\theta,t)d\theta, X_\rho(x,0) = \langle w_2, x\rangle \quad (3)$$

The ODE (3) is understood in the integrated sense, *i.e.*, for fixed distribution $\rho(\cdot,\cdot)$ and input $x \in \mathbb{R}^{d_1}$, the solution path $X_\rho(x,t), t \in [0,1]$ satisfies

$$X_\rho(x,t) = X_\rho(x,0) + \int_0^t \int_\Omega f(X_\rho(x,s),\theta)\rho(\theta,s)d\theta ds.$$

Here $y(x) = \mathbb{E}[y|x] \in \mathbb{R}$ is the function to be estimated. The parameter $w_2 \in \mathbb{R}^{d_1\times d_2}$ represents the first convolution layer in the ResNet (He et al., 2016a;b), which extracts feature before sending them to the residual blocks.

To simplify the analysis, we let w_2 to a predefined linear transformation (*i.e.* not training the first layer parameters w_2) with the technical assumption that $\min\{\sigma(w_2)\} \geq \sigma_1$ and $\max\{\sigma(w_2)\} \leq \sigma_2$, where $\sigma(w_2)$ denotes the set of singular values. We remark that this assumption is not unrealistic, for example (Oyallon et al., 2017) let w_2 be a predefined wavelet transform and still achieved the state-of-the-art result on several benchmark datasets. Here $f(\cdot, \theta)$ is the residual block with parameter θ that aims to learn a feature transformation from $\mathbb{R}^{d_2}$ to $\mathbb{R}^{d_2}$. For simplicity, we assume that the residual block is a two layer neural network, thus $f(x, \theta) = \sigma(\theta x), \theta \in \Omega \subset \mathbb{R}^{d_2 \times d_2}$ and $\sigma : \mathbb{R} \to \mathbb{R}$ is an activation function, such as sigmoid and relu. Note that in our notation $\sigma(\theta x)$ the activation function σ is applied separately to each component of the vector.

Finally, $w_1 \in \mathbb{R}^{d_2 \times 1}$ is a pooling operator that transfers the final feature $X_\rho(x, 1)$ to the classification result and an l_2 loss function is used for example. We also assume that w_1 is a predefined linear transform with satisfies $\|w_1\|_2 = 1$, which can be easily achieved via an operator used in realistic architecture such as the global average pooling (Lin et al., 2013). Before starting the analysis, we first list the necessary regularity assumptions.

Assumption 1.

1. *(Boundedness of data and target distribution) The input data x lies μ-almost surely in a compact ball,* i.e. *$\|x\| \leq R_1$ for some constant $R_1 > 0$. At the same time the target function is also bounded $\|y(\cdot)\|_\infty \leq R_2$ for some constant $R_2 > 0$.*
2. *(Lipschitz continuity of distribution with respect to depth) There exists a constant C_ρ such that*

   $$\|\rho(\cdot, t_1) - \rho(\cdot, t_2)\|_{BL} \leq C_\rho |t_1 - t_2|$$

   for all $t_1, t_2 \in [0, 1]$.
3. *The kernel $k(x_1, x_2) := g(x_1, x_2) = \sigma(x_1^\intercal x_2)$ is a universal kernel (Micchelli et al., 2006), i.e. the span of $\{k(x, \cdot) : x \in \mathbb{R}^{d_2}\}$ is dense in L^2.*
4. *(Locally Lipschitz derivative with sub-linear growth (Chizat & Bach, 2018)) There exists a family $\{Q_r\}_{r>0}$ of nested nonempty closed convex subsets of Ω that satisfies:*
   - *$\{u \in \Omega \mid \operatorname{dist}(u, Q_r) \leq r'\} \subset Q_{r+r'}$ for all $r, r' > 0$.*
   - *There exist constants $C_1, C_2 > 0$ such that*

     $$\sup_{\theta \in Q_r, x} \|\nabla_x f(x, \theta)\| \leq C_1 + C_2 r$$

     holds for all $r > 0$. Also the gradient of $f(x, \theta)$ with respect to x is a Lipschitz function with Lipschitz constant $L_r > 0$.
   - *For each r, the gradient respect to the parameter θ is also bounded*

     $$\sup_{\|x\| \leq R_1, \theta \in Q_r} \|\nabla_\theta f(x, \theta)\| \leq C_{3,r}$$

     for some constant $C_{3,r}$.

Remark. Let us elaborate on these assumptions in the neural network setting. For Assumption 1.4, $k(x_1, x_2) := g(x_1, x_2) = \sigma(x_1^\intercal x_2)$ is a universal kernel holds for the sigmoid and ReLU activation function. The local regularity Assumption 1.5 concerning function $f(x, \theta)$ can easily be satisfied, for $\nabla_\theta \sigma(\theta^\intercal x) = \sigma'(\theta^\intercal x)x$ and $\nabla_x \sigma(\theta^\intercal x) = \sigma'(\theta^\intercal x)\theta$. Hence, in order to satisfy the local regularity condition, one possible solution is that we utlize a Lipschitz gradient activation function and set the local set Q_r to be a ball with radius r centered at origin.

Under these assumptions, we can establish the existence, uniqueness, stability, and well-posedness of our forward model.

Theorem 1. *Under Assumption 1 and we further assume that there exist an $r > 0$ such that μ is concentrated on one of the nested sets Q_r. Then the ODE in (3) has a unique solution in $t \in [0, 1]$ for any initial condition $x \in \mathbb{R}^{d_1}$ with $\|x\| \leq R_1$. Moreover, for any pair of distributions ρ_1 and ρ_2, there exists a constant C such that*

$$\|X_{\rho_1}(x, 1) - X_{\rho_2}(x, 1)\| < CW_2(\rho_1, \rho_2), \tag{4}$$

for any $\|x\| \leq R_1$.

2.2. Deep Residual Network Behaves Like an Ensemble Of Shallow Models

In this section, we briefly explain the intuition behind our analysis, *i.e.* deep residual network can be approximated by a two-layer neural network. Veit et al. (2016) introduced an unraveled view of the ResNets and showed that deep ResNets behave like ensembles of shallow models. First, we offer a formal derivation to reveal how to make connection between a deep ResNet and a two-layer neural network. The first residual block is formulated as

$$X^1 = X^0 + \frac{1}{L}\int_{\theta^0} \sigma(\theta^0 X^0)\rho^0(\theta^0)d\theta^0.$$

By Taylor expansion, the second layer output is given by

$$\begin{aligned}
&X^2 = X^1 + \frac{1}{L}\int_{\theta^1} \sigma(\theta^1 X^1)\rho^1(\theta^1)d\theta^1 \\
&= X^0 + \frac{1}{L}\int_{\theta^0} \sigma(\theta^0 X^0)\rho^0(\theta^0)d\theta^0 \\
&+ \int_{\theta^1} \sigma(\theta^1(X^0 + \frac{1}{L}\int_{\theta^0} \sigma(\theta^0 X^0)\rho^0(\theta^0)d\theta^0))\rho^1(\theta^1)d\theta^1 \\
&= X^0 + \frac{1}{L}\int_{\theta^0} \sigma(\theta^0 X^0)\rho^0(\theta^0)d\theta^0 \\
&+ X^0 + \frac{1}{L}\int_{\theta^1} \sigma(\theta^1 X^0)\rho^1(\theta^1)d\theta^1 \\
&+ \frac{1}{L^2}\int_{\theta_1} \nabla\sigma(\theta^1 X^0)\theta^1(\int_{\theta^0} \sigma(\theta^0 X^0)\rho^0(\theta^0)d\theta^0)\rho^1(\theta^1)d\theta^1 \\
&+ h.o.t.
\end{aligned}$$

Iterating this expansion gives rise to

$$X^L \approx X^0 + \frac{1}{L}\sum_{a=0}^{L-1}\int \sigma(\theta X^0)\rho^a(\theta)d\theta$$
$$+ \frac{1}{L^2}\sum_{b>a}\int\int \nabla\sigma(\theta^b X^0)\theta^b\sigma(\theta^a X^0)\rho^b(\theta^b)\rho^a(\theta^a)d\theta^b\theta^a$$
$$+ h.o.t.$$

Here we only keep the terms that are at most quadratic in ρ. A similar derivation shows that at order k in ρ there are $\binom{L}{k}$ terms with coefficient $\frac{1}{L^k}$ each. This implies that the k-th order term in ρ decays as $O(\frac{1}{k!})$, suggesting that one can approximate a deep network by the keeping a few leading orders.

3. Landscape Analysis of the Mean-Field Model

In the following, we show that the landscape of a deep residual network enjoys the extraordinary property that any local optima is global, by comparing the gradient of deep residual network with the mean-field model of two-layer neural network (Mei et al., 2018; Chizat et al., 2019; Nitanda & Suzuki, 2017). To estimate the accuracy of the first order approximation (*i.e.* linearization), we apply the adjoint sensitivity analysis (Boltyanskiy et al., 1962) and show that the difference between the gradient of two models can be bounded via the stability constant of the backward adjoint equation. More precisely, the goal is to show the backward adjoint equation will only affect the gradient in a bounded constant.

3.1. Gradient via the Adjoint Sensitivity Method

Adjoint Equation. To optimize the objective (2), we calculate the gradient $\frac{\delta E}{\delta \rho}$ via the *adjoint sensitivity method* (Boltyanskiy et al., 1962). To derive the adjoint equation, we first view our generative models where ρ is treated as a parameter as

$$\dot{X}(x,t) = F(X(x,t);\rho), \tag{5}$$

with

$$F(X(x,t);\rho) = \int f(X(x,t);\theta)\rho(\theta,t)\,\mathrm{d}\theta. \tag{6}$$

The loss function can be written as

$$\mathbb{E}_{x\sim\mu}E(x;\rho) := \mathbb{E}_{x\sim\mu}\frac{1}{2}\left|\langle w_1, X_\rho(x,1)\rangle - y(x)\right|^2 \tag{7}$$

Define

$$p_\rho(x,1) := \frac{\partial E(x;\rho)}{\partial X_\rho(x,1)} = \left(\langle w_1, X_\rho(x,1)\rangle - y(x)\right)w_1 \tag{8}$$

The derivative of $X(x,1)$ with respect to $X(x,s)$, denoted by the Jacobian $J_\rho(x,s)$, satisfies at any previous time $s \le 1$ the adjoint equation of the ODE

$$\dot{J}_\rho(x,s) = -J_\rho(x,s)\nabla_X F(X_\rho(x,s);\rho). \tag{9}$$

Next, the perturbation of E by ρ is given by chain rule as

$$\begin{aligned}\frac{\delta E}{\delta\rho(s)} &= \frac{\partial E}{\partial X_\rho(X,1)}\frac{\delta X_\rho(x,1)}{\delta\rho(s)}\\ &= \frac{\partial E}{\partial X_\rho(X,1)}J_\rho(x,s)\frac{\delta F(X_\rho(x,s);\rho)}{\delta\rho(s)}\\ &= p_\rho(x,s)\, f(X_\rho(x,has),\cdot),\end{aligned} \tag{10}$$

where $p_\rho(x,s)$ (the derivative of $E(x;\rho)$ with respect to $X_\rho(x,s)$) satisfies the adjoint equation

$$\begin{aligned}\dot{p}_\rho(x,t) &= -\delta_X H_\rho(p_\rho,x,t)\\ &= -p_\rho(x,t)\int \nabla_X f(X_\rho(x,t),\theta)\rho(\theta,t)d\theta,\end{aligned}$$

which represents the gradient as a second backwards-in-time augmented ODE. Here the Hamiltonian is defined as $H_\rho(p,x,t) = p(x,t)\cdot\int f(x,\theta)\rho(\theta,t)d\theta$.

Utilizing the adjoint equation, we can characterize the gradient of our model with respect to the distribution ρ. More precisely, we may characterize the variation of the loss function with respect to the distribution as the following theorem. A detailed proof is presented in Appendix.

Theorem 2. *(Gradient of the parameter) For $\rho \in \mathcal{P}^2$ let*

$$\frac{\delta E}{\delta\rho}(\theta,t) = \mathbb{E}_{x\sim\mu}f(X_\rho(x,t),\theta))p_\rho(x,t).$$

Then for every $\nu \in \mathcal{P}^2$, we have

$$E(\rho + \lambda(\nu-\rho)) = E(\rho) + \lambda\left\langle\frac{\delta E}{\delta\rho},(\nu-\rho)\right\rangle + o(\lambda)$$

for the convex combination $(1-\lambda)\rho + \lambda\nu \in \mathcal{P}^2$ with $\lambda \in [0,1]$.

Remark. The adjoint equation, *i.e.*, the backward dynamical system, can be understood as a continuum limit of the back-propagation algorithm (LeCun et al., 1988; Li et al., 2017; Zhang et al., 2019a).

3.2. Landscape Analysis

In this section we aim to show that the proposed model enjoys a good landscape in the L_2 geometry. Specifically, we can always find a descent direction around a point whose loss is strictly larger than 0, which means that all local minimum is a global one. We list here a proof sketch and the details are given in the Appendix.

Theorem 3. *If $E(\rho) > 0$ for distribution $\rho \in \mathcal{P}^2$ that is supported on one of the nested sets Q_r, we can always construct a descend direction $\nu \in \mathcal{P}^2$,* i.e.

$$\inf_{\nu \in \mathcal{P}^2} \left\langle \frac{\delta E}{\delta \rho}, (\nu - \rho) \right\rangle < 0$$

Proof. First we lower bound the gradient with respect to the feature map $X_\rho(\cdot, t)$ by the loss function to show that changing feature map can always leads to a lower loss. This is also observed by Bartlett et al. (2018; 2019), where they proposed a functional derivative analysis of the compositional model of near-identity functions.

The next lemma aims to show that the backward adjoint process will not lose most of the information during propagating the gradient, which is the reason why we claim a ResNet's gradient is similar to a two-layer one.

Lemma 1. *The norm of the solution to the adjoint equation can be bounded by the loss*

$$\|p_\rho(\cdot, t)\|_\mu \geq e^{-(C_1 + C_2 r)} E(\rho), \forall t \in [0, 1]$$

where C_1 and C_2 are some constants and r is the same as in Theorem 3.

Then we follow the idea that the neural network is a linear model in $\mathcal{P}^2$ respect to the distribution of the weight(Bengio et al., 2006; Bach, 2017; Mei et al., 2018). Thanks to the existence and uniqueness of the solution of the ODE model as stated in Theorem 1, the solution map of the ODE is invertible so that there exists an inverse map $X_{\rho,t}^{-1}$ such that we can construct an inversion function $X_{\rho,t}^{-1}(X_\rho(x, t)) = x$. With $X_{\rho,t}^{-1}$, we define $\hat{p}_\rho(x, t) = p_\rho(X_{\rho,t}^{-1}(x), t)$.

Since $\rho(\theta, t)$ is a probability density, i.e., $\int \int \rho(\theta, t) d\theta dt = 1$, there exists $t_* \in (0, 1)$ such that $\int_\theta \rho(\theta, t_*) d\theta > \frac{1}{2}$. Since $k(x_1, x_2) = f(x_1, x_2)$ is a universal kernel (Micchelli et al., 2006), for any $g(x)$ satisfying that $\|g\|_{\hat{\mu}} < \infty$ for some probability measure $\hat{\mu}$ and for any fixed $\epsilon > 0$, there exists a probability distribution $\delta\hat{\nu} \in \mathcal{P}^2(\mathbb{R}^{d_2})$ such that

$$\|g(x) - \int_\theta f(x, \theta) \delta\hat{\nu}(\theta) d\theta\|_{\hat{\mu}} \leq \epsilon, \tag{11}$$

In particular, in what follows we consider the function $g(x)$ and the measure $\hat{\mu}$ given by

$$g(x) := -\hat{p}(x, t_*) + \frac{1}{\int_\theta \rho(\theta, t_*) d\theta} \int_\theta f(x, \theta) \rho(\theta, t_*) d\theta$$

where $\hat{\mu} = \hat{\mu}_{\rho, t_*} := X_\rho(\cdot, t_*)_{\#}\mu$. The value of ϵ will be chosen later in the proof. Moreover, we also define the perturbed measure

$$\delta\nu = \left(\delta\hat{\mu}(\theta) - \frac{\rho(\theta, t_*)}{\int_\theta \rho(\theta, t_*) d\theta} \right) \phi(t), \tag{12}$$

where $\phi(t)$ is a smooth non-negative function integrates to 1 and compactly supported in the interval $(0, 1)$, so that it is clear that $\delta\nu$ satisfies the regularity assumptions. We will consider the perturbed probability density ν defined as

$$\nu = \rho + \delta r \delta\nu \text{ for some } \delta r > 0.$$

Lemma 2. *The constructed ν with ϵ sufficiently small gives a descent direction of our model with the estimate*

$$\left\langle \frac{\delta E}{\delta \rho}, (\nu - \rho) \right\rangle \leq -\frac{\delta r}{2} e^{-2(C_1 + C_2 r)} E(\rho) < 0. \tag{13}$$

As Lemma 2 illustrated, if the loss $E(\rho)$ is not equal to zero, then we can always find a direction to decrease the loss, which proves Theorem 3. □

3.3. Discussion of the Wasserstein gradient flow

As described in the introduction, we consider each residual block as a particle and trace the evolution of the empirical distribution ρ_s of the particles during the training (here the variable s denotes the training time). While using gradient descent or stochastic gradient descent with small time steps, we move each particle through a velocity field $\{v_s\}_{s \geq 0}$ and the evolution can be expressed by a PDE $\partial_s \rho_s = \text{div}(\rho_s v_s)$, where div is the divergence operator. Several recent papers (Mei et al., 2018; Chizat & Bach, 2018; Rotskoff & Vanden-Eijnden, 2018) have shown that when the gradient field is gained from a (stochastic) gradient descent algorithm for training a particle realization of the mean-field model, the PDE is the Wasserstein gradient flow of the objective function. Thus in this section, we consider the gradient flow of the the objective function in the Wasserstein space, given by a McKean–Vlasov type equation (Carrillo et al., 2003; Ambrosio et al., 2008; Jordan et al., 1998; Otto, 2001; Nitanda & Suzuki, 2017)

$$\frac{\partial_{(\theta,t)} \rho}{\partial s} = \text{div}_{(\theta,t)} \left(\rho \nabla_{(\theta,,t)} \frac{\delta E}{\delta \rho} \right). \tag{14}$$

We consider the stationary point of such flow, *i.e.*, distribution ρ such that the right hand side is 0. Our next result shows that such stationary points are global minimum of the loss function under the homogeneous assumption of the residual block and a separation property of the support of the stationary distribution.

Theorem 4. *(Informal) When the residual block (X, θ) is positively p-homogeneous respective to θ. Let $(\rho_s)_{s \geq 0}$ be the solution of the the Wasserstein gradient $\frac{\partial_{(\theta,t)} \rho}{\partial s} = div_{(\rho,t)}(\rho \nabla_{(\rho,t)} \frac{\delta E}{\delta \rho})$ of our mean-field model (3). If $(\rho_s)_{s \geq 0}$ converge to ρ_∞ in W_2 and ρ^* concentrates in a ball $B(0, r_b)$ and separates the spheres $r_a \mathbb{S}^{d-1} \times [0, 1]$ and $r_b \mathbb{S}^{d-1} \times [0, 1]$. Then ρ^* is the global minimum satisfies $E(\rho_\infty) = 0$.*

The precise statement and the proof are presented in Appendix, where we also analyze the regularity of our objective function in the Wasserstein space. The result guarantees that when the gradient flow converges, it has to reach the global minimum of the loss function.

4. Deep ResNet as Numerical Scheme

In this section, following (Bengio et al., 2006; Lu et al., 2017), we aim to design scalable deep learning algorithms via the discretization of the continuous model. We use a set of particles to approximate the the distribution (Nitanda & Suzuki, 2017; Ba et al., 2019; Liu & Wang, 2016) and Euler scheme to numerical solve the ODE model which leads to a simple Residual Network (Lu et al., 2017).

To simulate the Wasserstein gradient flow (14) via a stochastic gradient descent algorithm, we use a particle representation of the distribution $\rho(x,t)$, commonly used in the literature, see e.g., (Liu & Wang, 2016; Nitanda & Suzuki, 2017; Rotskoff et al., 2019; Mei et al., 2018; Chizat & Bach, 2018). In the two-layer neural network, the particle realization becomes the standard training procedure of using (stochastic) gradient descent. Our aim is to extend this approach to deep residual networks, starting from the continuum mean-field model presented above. Since ρ characterizes the distribution of the pairs (θ, t), each particle in our representation would carry the parameter θ, together with information on the activation time period of the particle. Therefore, also different from the usual standard ResNet, we also need to allow the particle to move in the gradient direction corresponding to t. We may consider using a parametrization of ρ with n particles as

$$\rho_n(\theta,t) = \sum_{i=1}^{n} \delta_{\theta_i}(\theta)\mathbb{1}_{[\tau_i,\tau_i']}(t).$$

The characteristic function $\mathbb{1}_{[\tau_i,\tau_i']}$ can be viewed as a relaxation of the Dirac delta mass $\delta_{\tau_i}(t)$. However, this parametrization comes with a difficulty in practice, namely, the intervals $[t_i, t_i']$ may overlap significantly with each other, and in the worst case, though unlikely, all the time intervals of the n particles coincide, which leads to heavy computational cost in the training process.

Therefore, for practical implementation, we constrain that every time instance t is just contained in the time interval of a single particle. We realize this by adding a constraint $\tau_i' = \tau_{i+1}$ between consecutive intervals. More precisely, given a set of parameters (θ^i, τ^i), we first sort them according to τ^i values. Assuming τ^i are ordered, we define the architecture as

$$X^{\ell+1} = X^\ell + (\tau^\ell - \tau^{\ell-1})\sigma(\theta^\ell X^\ell), \quad 0 \le \ell < n; \tag{15}$$

$$X^0 = x. \tag{16}$$

Both θ and τ parameters can be trained with SGD and n is the depth of the network. The order of τ may change during the training (thus to make each particle indistinguishable to guarantee the mean-field behavior), thus after every update, we *sort* the τ_i to get the new order of the residual blocks. The algorithm is listed in Algorithm 1. The new algorithm only introduces n parameters, as n is the depth which is around 100 in practice, thus the number of extra parameters is negligible comparing to the 1M+ parameter number typically used in usual ResNet architectures. The sorting of $\{\tau_i\}_{i=1}^n$ also induces negligible cost per step.

We also remark that the flexibility of τ^ℓ can be also viewed as an adaptive time marching scheme of the ODE model for x, as $\tau^\ell - \tau^{\ell-1}$ can be understood as the time step in the Euler discretization. Since the parameters $\{\tau^\ell\}$ are learned from data, as a by-product, our scheme also naturally yields a data-adaptive discretization scheme.

Algorithm 1 Training Of Mean-Field Deep Residual Network

```
  Given: A collection of residual blocks $(\theta_i, \tau_i)_{i=1}^n$
  while training do
    Sort $(\theta_i, \tau_i)$ based on $\tau_i$ to be $(\theta^i, \tau^i)$ where $\tau^0 \le \cdots \le \tau^n$.
    Define the ResNet as $X^{\ell+1} = X^\ell + (\tau^\ell - \tau^{\ell-1})\sigma(\theta^\ell X^\ell)$ for $0 \le \ell < n$.
    Use gradient descent to update both $\theta^i$ and $\tau^i$.
  end while
```

As the number of particles n becomes large, the expected time evolution of ρ_n should be close to the gradient flow (14). The rigorous proof of this is however non-trivial, which will be left for future works.

5. Experiment

In this section, we aim to show that our algorithm is not only designed from theoretical consideration but also realizable on practical datasets and network structures. We implement our algorithm for ResNet/ResNeXt on CIFAR 10/100 datasets and demonstrate that our "mean-field training" method consistently outperforms the vanilla stochastic gradient descent.

Implementation Details.

On CIFAR, we follow the simple data augmentation method in (He et al., 2016a;b) for training: 4 pixels are padded on each side, and a 32×32 crop is randomly sampled from the padded image or its horizontal flip. For testing, we only evaluate the single view of the original 32×32 image.

For the experiments of ResNet on CIFAR, we adopt the original design of the residual block in He et al. (2016a), i.e.

using a small two-layer neural network as the residual block, whose layered structure is bn-relu-conv-bn-relu-conv. We start our networks with a single 3×3 conv layer, followed by 3 residual blocks, a global average pooling, and a fully-connected classifier. Parameters are initialized following the method introduced by He et al. (2015). Mini-batch SGD is used to optimize the parameters with a batch size of 128. During training, we apply a weight decay of 0.0001 for ResNet and 0.0005 for ResNeXt, and a momentum of 0.9.

For ResNet on CIFAR10 (CIFAR100), we start with the learning rate of 0.1, divide it by 10 at 80 (150) and 120 (225) epochs and terminate the training at 160 (300) epochs. For ResNeXt on CIFAR100, we start with the learning rate of 0.1 and divide it by 10 at 150 and 225 epochs, and terminate the training at 300 epochs. We would like to mention that here the ResNeXt is a preact version which is different from the original (Xie et al., 2017). This difference leads to a small performance drop on the final result. For each model and dataset, we report the average test accuracy over 3 runs in Table 5.

6. Discussion and Conclusion

6.1. Conclusion

To better understand the reason that stochastic gradient descent can optimize the complicated landscape. Our work directly consider an infinitely deep residual network. We proposed a new continuous model of deep ResNets and established an asymptotic global optimality property by bounding the difference between the gradient of the deep residual network and an associated two-layer network. Our analysis can be considered as a theoretical characterization of the observation that a deep residual network looks like a shallow model ensemble (Veit et al., 2016) by utilizing ODE and control theory. Based on the new continuous model, we consider the original residual network as an approximation of the continuous model and proposed a new training method. The new method involves a step of sorting residual blocks, which introduces essentially no extra computational effort but results in better empirical results.

	Vanilla	mean-field	Dataset
ResNet20	8.75	8.19	CIFAR10
ResNet32	7.51	7.15	CIFAR10
ResNet44	7.17	6.91	CIFAR10
ResNet56	6.97	6.72	CIFAR10
ResNet110	6.37	6.10	CIFAR10
ResNet164	5.46	5.19	CIFAR10
ResNeXt29(864d)	17.92	17.53	CIFAR100
ResNeXt29(1664d)	17.65	16.81	CIFAR100

Table 1. Comparison of the stochastic gradient descent and mean-field training (Algorithm 1.) of ResNet On CIFAR Dataset. Results indicate that our method our performs the Vanilla SGD consistently.

6.2. Discussion and Future Work

Our work gives qualitative analysis of the loss landscape of a deep residual network and shows that its gradient differs from the gradient of a two-layer neural network by at most a bounded factor when the loss is at the same level. This indicates that the deep residual network's landscape may not be much more complicate than a two-layer network, which inspires us to formulate a mean-field analysis framework for deep residual network and suggests a possible framework for the optimization of the deep networks beyond the kernel regime. (Yun et al., 2019) has shown that deep residual network may not be better than a linear model in terms of optimization, but our work suggests that this is caused by the lack of overparameterization. In the highly overparameterization regime, the landscape of deep ResNet can still be nice. Based on the initiation and framework proposed in our paper, there are several interesting directions related to understanding and improving the residual networks.

Firstly, to ensure the full support assumption, we can consider extending the neural birth-death (Rotskoff et al., 2019; Chizat, 2019) to deep ResNets. Neural birth-death dynamics considers the gradient flow in the Wasserstein-Fisher-Rao space(Chizat et al., 2018) rather than the Wasserstein space and ensures convergence. It's also interesting to extend the convergence proof to other optimization algorithms like natrual gradient descent (Kingma & Ba, 2014; Amari et al., 2020; Wu & Xu, 2020).

Secondly, as shown in the derivation in Section 2.2, the two-layer network approximation is just the lowest order approximation to the deep residual network and it is interesting to explore the higher order terms.

Acknowledgments

We thank the hospitality of the American Institute of Mathematics (AIM) for the workshop "Deep learning and partial differential equation" in October 2019, which led to this collaborative effort. Yiping Lu also thanks Denny Wu, Xuechen Li and Pingchuan Ma for helpful comments and feedback. The work of Jianfeng Lu is supported in part by the National Science Foundation via grants DMS-1454939 and CCF-1934964 (Duke TRIPODS). The work of Lexing Ying is supported in part by the National Science Foundation via grant DMS-1818449.

References

Allen-Zhu, Z. and Li, Y. What can resnet learn efficiently, going beyond kernels? In *Advances in Neural Information Processing Systems*, pp. 9015–9025, 2019.

Allen-Zhu, Z., Li, Y., and Song, Z. A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*, 2018.

Amari, S.-i., Ba, J., Grosse, R., Li, X., Nitanda, A., Suzuki, T., Wu, D., and Xu, J. When does preconditioning help or hurt generalization? *arXiv preprint arXiv:2006.10732*, 2020.

Ambrosio, L., Gigli, N., and Savaré, G. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2008.

Araújo, D., Oliveira, R. I., and Yukimura, D. A mean-field limit for certain deep neural networks. *arXiv preprint arXiv:1906.00193*, 2019.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R., and Wang, R. On exact computation with an infinitely wide neural net. *arXiv preprint arXiv:1904.11955*, 2019.

Avelin, B. and Nyström, K. Neural odes as the deep limit of resnets with constant weights. *arXiv preprint arXiv:1906.12183*, 2019.

Ba, J., Erdogdu, M. A., Ghassemi, M., Suzuki, T., Sun, S., Wu, D., and Zhang, T. Towards characterizing the high-dimensional bias of kernel-based particle inference algorithms. 2019.

Ba, J., Erdogdu, M., Suzuki, T., Wu, D., and Zhang, T. Generalization of two-layer neural networks: An asymptotic viewpoint. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1gBsgBYwH.

Bach, F. Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research*, 18(1):629–681, 2017.

Bartlett, P. L., Evans, S. N., and Long, P. M. Representing smooth functions as compositions of near-identity functions with implications for deep network optimization. *arXiv preprint arXiv:1804.05012*, 2018.

Bartlett, P. L., Helmbold, D. P., and Long, P. M. Gradient descent with identity initialization efficiently learns positive-definite linear transformations by deep residual networks. *Neural computation*, 31(3):477–502, 2019.

Bengio, Y., Roux, N. L., Vincent, P., Delalleau, O., and Marcotte, P. Convex neural networks. In *Advances in neural information processing systems*, pp. 123–130, 2006.

Boltyanskiy, V., Gamkrelidze, R., MISHCHENKO, Y., and Pontryagin, L. Mathematical theory of optimal processes. 1962.

Carrillo, J. A., McCann, R. J., Villani, C., et al. Kinetic equilibration rates for granular media and related equations: entropy dissipation and mass transportation estimates. *Revista Matematica Iberoamericana*, 19(3):971–1018, 2003.

Chen, T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, pp. 6572–6583, 2018.

Chizat, L. Sparse optimization on measures with over-parameterized gradient descent. *arXiv preprint arXiv:1907.10300*, 2019.

Chizat, L. and Bach, F. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in neural information processing systems*, pp. 3036–3046, 2018.

Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F.-X. An interpolating distance between optimal transport and fisher–rao metrics. *Foundations of Computational Mathematics*, 18(1):1–44, 2018.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. 2019.

Du, S. S., Lee, J. D., Li, H., Wang, L., and Zhai, X. Gradient descent finds global minima of deep neural networks. *arXiv preprint arXiv:1811.03804*, 2018.

Dupont, E., Doucet, A., and Teh, Y. W. Augmented neural odes. In *Advances in Neural Information Processing Systems*, pp. 3134–3144, 2019.

E, W. A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, 5 (1):1–11, 2017.

E, W., Han, J., and Li, Q. A mean-field optimal control formulation of deep learning. *Research in the Mathematical Sciences*, 6(1):10, 2019a.

E, W., Ma, C., and Wu, L. Machine learning from a continuous viewpoint, 2019b.

Fang, C., Gu, Y., Zhang, W., and Zhang, T. Convex formulation of overparameterized deep neural networks, 2019.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. Limitations of lazy training of two-layers neural networks. *arXiv preprint arXiv:1906.08899*, 2019.

Haber, E. and Ruthotto, L. Stable architectures for deep neural networks. *Inverse Problems*, 34(1):014004, 2017.

Hardt, M. and Ma, T. Identity matters in deep learning. *arXiv preprint arXiv:1611.04231*, 2016.

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

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016a.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *European conference on computer vision*, pp. 630–645. Springer, 2016b.

Hu, K., Ren, Z., Siska, D., and Szpruch, L. Mean-field langevin dynamics and energy landscape of neural networks. *arXiv preprint arXiv:1905.07769*, 2019.

Ishida, T., Yamane, I., Sakai, T., Niu, G., and Sugiyama, M. Do we need zero training loss after achieving zero training error? *arXiv preprint arXiv:2002.08709*, 2020.

Jabir, J.-F., Šiška, D., and Szpruch, Ł. Mean-field neural odes via relaxed optimal control. *arXiv preprint arXiv:1912.05475*, 2019.

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

Jordan, R., Kinderlehrer, D., and Otto, F. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

Kawaguchi, K. and Bengio, Y. Depth with nonlinearity creates no bad local minima in resnets. *Neural Networks*, 118:167–174, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

LeCun, Y., Touresky, D., Hinton, G., and Sejnowski, T. A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, volume 1, pp. 21–28. CMU, Pittsburgh, Pa: Morgan Kaufmann, 1988.

Li, Q. and Hao, S. An optimal control approach to deep learning and applications to discrete-weight neural networks. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2985–2994, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/li18b.html.

Li, Q., Chen, L., Tai, C., and E, W. Maximum principle based algorithms for deep learning. *The Journal of Machine Learning Research*, 18(1):5998–6026, 2017.

Li, X., Wong, T.-K. L., Chen, R. T., and Duvenaud, D. Scalable gradients for stochastic differential equations. *arXiv preprint arXiv:2001.01328*, 2020.

Li, Y. and Yuan, Y. Convergence analysis of two-layer neural networks with relu activation. In *Advances in Neural Information Processing Systems*, pp. 597–607, 2017.

Lin, M., Chen, Q., and Yan, S. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

Liu, G.-H. and Theodorou, E. A. Deep learning theory review: An optimal control and dynamical systems perspective. *arXiv preprint arXiv:1908.10920*, 2019.

Liu, Q. and Wang, D. Stein variational gradient descent: A general purpose bayesian inference algorithm. In *Advances in neural information processing systems*, pp. 2378–2386, 2016.

Liu, T., Chen, M., Zhou, M., Du, S. S., Zhou, E., and Zhao, T. Towards understanding the importance of shortcut connections in residual networks. In *Advances in Neural Information Processing Systems*, pp. 7890–7900, 2019.

Lu, Y., Zhong, A., Li, Q., and Dong, B. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. *arXiv preprint arXiv:1710.10121*, 2017.

Lu, Y., Li, Z., He, D., Sun, Z., Dong, B., Qin, T., Wang, L., and Liu, T.-Y. Understanding and improving transformer from a multi-particle dynamic system point of view. *arXiv preprint arXiv:1906.02762*, 2019.

Ma, C., Wang, Q., et al. A priori estimates of the population risk for residual networks. *arXiv preprint arXiv:1903.02154*, 2019.

Mei, S., Montanari, A., and Nguyen, P.-M. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33): E7665–E7671, 2018.

Micchelli, C. A., Xu, Y., and Zhang, H. Universal kernels. *Journal of Machine Learning Research*, 7(Dec):2651–2667, 2006.

Nguyen, P.-M. Mean field limit of the learning dynamics of multilayer neural networks. *arXiv preprint arXiv:1902.02880*, 2019.

Nitanda, A. and Suzuki, T. Stochastic particle gradient descent for infinite ensembles. *arXiv preprint arXiv:1712.05438*, 2017.

Otto, F. The geometry of dissipative evolution equations: the porous medium equation. 2001.

Oyallon, E., Belilovsky, E., and Zagoruyko, S. Scaling the scattering transform: Deep hybrid networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 5618–5627, 2017.

Oymak, S. and Soltanolkotabi, M. Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *arXiv preprint arXiv:1902.04674*, 2019.

Rotskoff, G., Jelassi, S., Bruna, J., and Vanden-Eijnden, E. Neuron birth-death dynamics accelerates gradient descent and converges asymptotically. In *International Conference on Machine Learning*, pp. 5508–5517, 2019.

Rotskoff, G. M. and Vanden-Eijnden, E. Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error. *arXiv preprint arXiv:1805.00915*, 2018.

Shamir, O. Are resnets provably better than linear predictors? In *Advances in neural information processing systems*, pp. 507–516, 2018.

Sirignano, J. and Spiliopoulos, K. Mean field analysis of neural networks: A central limit theorem. *Stochastic Processes and their Applications*, 2019.

Sonoda, S. and Murata, N. Double continuum limit of deep neural networks. In *ICML Workshop Principled Approaches to Deep Learning*, 2017.

Sonoda, S. and Murata, N. Transport analysis of infinitely deep neural network. *The Journal of Machine Learning Research*, 20(1):31–82, 2019.

Suzuki, T. Adaptivity of deep relu network for learning in besov and mixed smooth besov spaces: optimal rate and curse of dimensionality. *arXiv preprint arXiv:1810.08033*, 2018.

Thorpe, M. and van Gennip, Y. Deep limits of residual neural networks. *arXiv preprint arXiv:1810.11741*, 2018.

Veit, A., Wilber, M. J., and Belongie, S. Residual networks behave like ensembles of relatively shallow networks. In *Advances in neural information processing systems*, pp. 550–558, 2016.

Wei, C., Lee, J. D., Liu, Q., and Ma, T. Regularization matters: Generalization and optimization of neural nets vs their induced kernel. In *Advances in Neural Information Processing Systems*, pp. 9709–9721, 2019.

Wu, D. and Xu, J. On the optimal weighted ℓ_2 regularization in overparameterized linear regression. *arXiv preprint arXiv:2006.05800*, 2020.

Wu, L., Wang, Q., and Ma, C. Global convergence of gradient descent for deep linear residual networks. In *Advances in Neural Information Processing Systems*, pp. 13368–13377, 2019.

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

Yun, C., Sra, S., and Jadbabaie, A. Small nonlinearities in activation functions create bad local minima in neural networks. *arXiv preprint arXiv:1802.03487*, 2018.

Yun, C., Sra, S., and Jadbabaie, A. Are deep resnets provably better than linear predictors? In *Advances in Neural Information Processing Systems*, pp. 15660–15669, 2019.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Zhang, D., Zhang, T., Lu, Y., Zhu, Z., and Dong, B. You only propagate once: Painless adversarial training using maximal principle. *arXiv preprint arXiv:1905.00877*, 2019a.

Zhang, H., Yu, D., Chen, W., and Liu, T.-Y. Training over-parameterized deep resnet is almost as easy as training a two-layer network. *arXiv preprint arXiv:1903.07120*, 2019b.

Zhang, X., Lu, Y., Liu, J., and Dong, B. Dynamically unfolding recurrent restorer: A moving endpoint control method for image restoration. In *International Conference on Learning Representations*, 2019c. URL `https://openreview.net/forum?id=SJfZKiC5FX`.

Zou, D., Cao, Y., Zhou, D., and Gu, Q. Stochastic gradient descent optimizes over-parameterized deep relu networks. *arXiv preprint arXiv:1811.08888*, 2018.