Asymptotic analysis of Deep Learning algorithms

Alain Rossier

St Hugh's College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2023

To my best friend, Arnaud.

Acknowledgements

I am not a man of many words. This time however, I have to give credit where credit is due. J'ai eu l'immense chance de grandir au côté d'un autre passionné des mathématiques. Ensemble, nous avons entretenu une amitié profonde et une compétition saine, nous emmenant aux quatre coins du monde physique et mathématique. Sans toi Arnaud, je ne serai pas en train d'écrire ces mots.

Je suis aussi éternellement reconnaissant envers mes parents Philippe et Manuela, qui ont su me guider avec balance sur ce long chemin. Vous m'avez constamment poussé à l'excellence, tout en veillant à ce que je garde les pieds sur terre.

This thesis has forged many friendships, in particular with my fellow DPhil students Jonathan and Felix. You have been a constant source of happiness in Oxford with your banter, through our ups and downs. For all the memories we have shared, with Regan as well, I would embark on this journey all over again. Merci Camille pour avoir partagé un bout de la route avec moi, pour tes multiples relectures, et surtout pour ton soutien indéfectible dans les moments difficiles.

This work is the result of several collaborations over the years, and special thoughts go to Renyuan. You have been a continuous source of inspiration since the start of my DPhil, and you have been kind enough to invite me to USC, where I could deeply focus (and enjoy the sun).

L'origine de ce project de doctorat vient de Alain-Sam, qui a été non seulement un collaborateur exceptionnel, mais surtout un ami proche, toujours prêt à déconner dans les meilleurs moments. As a result, I would like to thank Instadeep and Karim for sponsoring my DPhil, and for giving me valuable advice and feedback.

Finally, I am greatly indebted to my supervisor, Rama. You have taught me how to ask the right questions, how to stay humble but ambitious, and curious but inquisitive. Your willingness to challenge the status quo, your rigorous standards and your god-like intuition have greatly contributed to my growth as a researcher, and I shall never forget that.

Abstract

We investigate the asymptotic properties of deep residual networks as the number of layers increases. We first show the existence of scaling regimes for trained weights markedly different from those implicitly assumed in the neural ODE literature. We study the convergence of the hidden state dynamics in these scaling regimes, showing that one may obtain an ODE, a stochastic differential equation (SDE) or neither. Furthermore, we derive the corresponding scaling limits for the backpropagation dynamics. Finally, we prove that in the case of a smooth activation function, the scaling regime arises as a consequence of using gradient descent. In particular, we prove linear convergence of gradient descent to a global minimum for the training of deep residual networks. We also show that if the trained weights, as a function of the layer index, admit a scaling limit as the depth increases, then the limit has finite $p-$variation with $p = 2$.

This work also investigate the mean-field limit of path-homogeneous neural architectures. We prove convergence of the Wasserstein gradient flow to a global minimum, and we derive a generalization bound based on the stability of the optimization algorithm for 2-layer neural networks with ReLU activation.

Contents

1 Introduction **1**

1.1 Learning theory framwework 1

1.1.1 Approximation 3

1.1.2 Optimization 5

1.1.3 Generalization 7

1.2 Asymptotics 11

1.2.1 Width goes to infinity 11

1.2.2 Depth goes to infinity 15

1.2.3 Learning rate goes to zero 17

2 Linear residual networks **20**

2.1 Recent work 20

2.2 Problem formulation 21

2.3 Description of the solutions 22

2.4 Global convergence and scaling 24

2.5 Existence of a scaling limit as the depth increases and connection to linear neural ODE 29

2.6 Numerical examples 30

3 Scaling properties of deep residual networks **33**

3.1 Introduction 33

3.2 Scaling regimes 35

3.2.1 Scaling regimes for trained network weights 35

3.2.2 Smoothness of weights with respect to the layer 37

3.3 Scaling behavior of trained weights: numerical experiments 38

3.3.1 Methodology 38

3.3.2 Results for fully-connected layers 39

3.3.3 Results for convolutional networks 44

3.3.4 Summary: three scaling regimes 47
3.4 Deep network limit 47
3.4.1 Scaling regime 1: ODE limit 47
3.4.2 Scaling regime 2 49
3.4.3 Link with numerical experiments 53
3.4.4 Detailed proofs 54
3.5 Asymptotic analysis of the backpropagation dynamics 69
3.5.1 Backpropagation in supervised learning 69
3.5.2 Backward equation for the Jacobian under Scaling regime 1 . 70
3.5.3 Backward equation for the Jacobian under Scaling regime 2 . 71
3.5.4 Proofs 73

4 Convergence and implicit regularisation of gradient descent for deep residual networks 83
4.1 Introduction 83
4.1.1 Convergence and regularization properties of deep learning algorithms 84
4.1.2 Contributions 85
4.2 Residual networks 86
4.3 Dynamics of weights and hidden states under gradient descent 88
4.3.1 Bounds on the hidden states, their Jacobians, and the loss gradients 89
4.3.2 Behaviour of weight norms along the gradient descent path . . 93
4.3.3 Local convergence of gradient descent 93
4.3.4 Scaling limit of trained weights 98
4.4 Numerical experiments 99
4.4.1 Identification of scaling behavior 99
4.4.2 Rate of convergence 100
4.4.3 Emergence of regularity of weights as a function of the layer index 100
4.5 Conclusion 102

5 Mean-field limit and global convergence of gradient descent for path-homogeneous models 103
5.1 Introduction 103
5.1.1 Outlook and contributions 104
5.1.2 Related work 105
5.1.3 Notation 107

5.2 Path-homogeneity 107
5.2.1 Definitions 107
5.2.2 Path-homogeneous function 107
5.2.3 Multi-layer ReLU networks 108
5.3 Global convergence of Wasserstein gradient flow for path-homogeneous models 109
5.3.1 Calculus on the space of measures 111
5.3.2 Particle and Wasserstein gradient flow 111
5.3.3 Convergence to the global minimum 116
5.3.4 Global convergence for continuous-depth residual networks . . 118
5.4 Generalization properties in the 2-homogeneous case 120
5.4.1 Assumptions and definitions 120
5.4.2 Preliminary analysis 122
5.4.3 Generalization bound: main result 124
5.5 Numerical experiments 127
5.5.1 Particle complexity for convergence of multi-layer ReLU networks 127
5.5.2 Convergence of the particle gradient flow of deep convolutional networks 128

Bibliography **129**

A Hyperparameters **146**

B Technical results of Chapter 4 **147**
B.1 Gradient of the loss function with respect to parameters 147
B.2 Boundedness of hidden states and Jacobians 148
B.3 Upper bounds on the gradient and Hessian of the loss function 149
B.4 Lower bounds on loss gradients 151
B.5 Weight norms and loss function under gradient descent 152
B.6 Supporting lemma for Theorem 4.6 156

C Technical results of Chapter 5 **158**
C.1 Properties of path-homogeneous functions 158
C.2 Auxiliary results and proofs of Section 5.3 161
C.2.1 Bound on the variation of the subgradient 161
C.2.2 Proof of Proposition 5.11 162
C.2.3 Bound on the evolution of gradient flow functionals 165

C.3 Proofs of Section 5.4 166
C.3.1 Proof of Lemma 5.20 166
C.3.2 Proof of Lemma 5.21 167

D Auxiliary results **169**

List of Figures

2.1 Frobenius norm of the trained weights at different depths with the ZAS initialization (2.5). In red: norm of the last layer. In blue: norms of all the other layers. 31

2.2 Learning rate $\eta^{(L)}$ used in SGD to train the weights at different depths with the ZAS initialization (2.5). The learning rate is updated according to the schedule described in Section 2.6. 31

2.3 Frobenius norm of the trained weights at different depths with the Xavier initialization. In red: norm of the last layer. In blue: norms of all the other layers. 32

2.4 Learning rate $\eta^{(L)}$ used in SGD to train the weights at different depths with the Xavier initialization. The learning rate is updated according to the schedule described in Section 2.6. 32

3.1 Trained weights as a function of k/L for $k = 0, \ldots, L$ and $L = 9100$. Left: rescaled weights $L^\beta A^{(L)}_{k,(0,0)}$ for a tanh network with $\beta = 0.2$. Right: cumulative sum $\sum_{j=0}^{k-1} A^{(L)}_{j,(0,0)}$for a ReLU network. Note that each $A^{(L)}_{k,(0,0)} \in \mathbb{R}$. 36

3.2 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. 41

3.3 Identification of scaling behavior in the case of tanh activation and $\delta^{(L)} \in \mathbb{R}$. Left: log-log plot of root sum of squares of $A^{(L)}$ (pink) and the β-scaled norm of increments of $A^{(L)}$ (orange). 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 (3.6), for the synthetic dataset. . . . 41

3.4 Scaling for ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$. Left: Cumulative sum norm of $|\delta^{(L)}|A^{(L)}$ with respect to L, in log-log scale. Right: trained values of $\delta_k^{(L)}$ as a function of k, for $L = 9100$ and for the synthetic dataset. 42

3.5 ReLU activation and scalar $\delta_k^{(L)}$. 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 $|\delta^{(L)}|\, A_{k,(7,7)}^{(L)}$ with the trend part $\overline{A}$ and the noise part W^A for $L = 10321$, as defined in (3.6), for the synthetic dataset. 42

3.6 Scaling behavior for $b^{(L)}$ with tanh activation and scalar $\delta^{(L)}$. Left: cumulative sum norm of $b^{(L)}$ with respect to L, in log-log scale. Middle: the root sum of squares of $b^{(L)}$ in pink and the $\beta-$scaled norm of increments of $b^{(L)}$ in orange, in log-log scale. The dashed lines are for the synthetic data and the solid lines are for MNIST. Right: Decomposition of the trained weights $b_{k,5}^{(L)}$ with the trend part $\overline{b}$ and the noise part W^b for $L = 10321$, as defined in (3.6), for the synthetic dataset. 43

3.7 Scaling and hypothesis verification for $b^{(L)}$ with ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$. Left: cumulative sum norm of $|\delta^{(L)}|b^{(L)}$ with respect to L, in log-log scale. Middle: the root sum of squares of $|\delta^{(L)}|b^{(L)}$ in pink and the $\beta-$scaled norm of increments of $|\delta^{(L)}|b^{(L)}$ in orange, in log-log scale. The dashed lines are for the synthetic data and the solid lines for MNIST. Right: Decomposition of the trained weights $|\delta^{(L)}|\, b_{k,6}^{(L)}$ with the trend part $\overline{b}$ and the noise part W^b for $L = 10321$, as defined in (3.6), for the synthetic dataset. 43

3.8 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. 44

3.9 Residual architecture. There are 4 blocks that are respectively repeated n_1, n_2, n_3 and n_4 times. The network depth is $L = n_1 + n_2 + n_3 + n_4$. The Basic Block architecture is detailed in Figure 3.10. 45

3.10 Basic Block from Figure 3.9. See (3.9) for details. 45

3.11 Scaling of $\Delta^{(L)}$ (left) and $A^{(L)}$ (right) against the network depth L for convolutional architectures on CIFAR-10. In blue: spectral norm of the kernels $\Delta_k^{(L)}$, resp. $A_k^{(L)}$, for $k = 0, \ldots, L-1$. In red: maximum norm, defined in Table 3.1. The plots are in log-log scale. 46

3.12 Scaling behavior of $\Delta^{(L)}$ (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 3.1. We obtain α and β from Figure 3.11. 46

4.1 Left: scaling α_t of δ_L against the initial scaling α_0 for different training times. Right: Average loss value across depths $L \in \left\{2^k : k \in [3, 12]\right\}$ for different initializations α_0, as a function of the number of gradient steps t. 99

4.2 Both figures: horizontal axis is the initial scaling β_0 of the weights A, and the vertical axis is the fixed scaling α_0 of δ_L. Left: Final total scaling $\alpha_0 + \beta_T$. Right: Average final loss after $T = 200$ epochs. The depths at which we train our networks are $L \in \left\{2^k : k \in [3, 10]\right\}$. . . . 100

4.3 Both figures: horizontal axis is the inverse loss level $1/\epsilon$, in log-scale, and the vertical axis is the number of gradient steps needed for the average loss to drop below ϵ. The average is taken over the depths $L \in \left\{2^k : k \in [3, 10]\right\}$. Left: constant learning rates $\eta_L(t) = \eta_0$. Right: decaying learning rates $\eta_L(t) = \eta_0(t+1)^{-1}$. 101

4.4 Evolution of weight norms along gradient descent path for different depths $L \in \{2^4, 2^5, 2^6, 2^8, 2^{10}\}$. Left: L^2-type norm $\overline{f}^{(L)}(t)$ as a function of gradient iterations. Right: Quadratic variation-type norm $\bar{g}^{(L)}(t)$ as a function of gradient iterations. 101

4.5 Scatter plot of the rescaled weights $L^{1/2} A_{k,(7,18)}^{(L)}(T)$ for different values of $L \in \{4^x : x \in [3, 6]\}$ at the end of the training $T = 500$. Horizontal axis is the scaled layer index k/L. 102

5.1 Neural network representation for the model $x \mapsto y = \overline{\Phi}(\mu_m)(x) = \int_\Theta \Phi(\theta)(x) \mathrm{d}\mu_m(\theta)$. Here, x is the input, $\Phi(\theta)$ is a neural network mapping with parameter θ, $\mu_m = \frac{1}{m}\sum_{i=1}^m \delta_{\theta_i}$ is an atomic measure over the parameter space with $\theta_1, \ldots, \theta_m \in \Theta$, $y_i = \Phi(\theta_i)(x)$ is the output of the neural network $\Phi(\theta_i)$, and $y = \frac{1}{m}\sum_{i=1}^m y_i$ is the output of the model.110

5.2 Final training loss against the number of particles m, with $m_0 = 6$. The light blue points are 10 individual runs, the solid blue line is their median and the red curve is only optimizing the last layer. Left: $d_0 = 64$, $K = 2$. Middle: $d_0 = 8$, $d_h = 8$, $K = 3$. Right: $d_0 = 4$, $d_h = 4$, $K = 4$. 127

5.3 Particle gradient flow on AlexNetSmall and VGG11 tested on CIFAR-10. Left: AlexNetSmall on CIFAR-10. Right: VGG11 (11 layers) 128

C.1 Illustration of the case $\alpha = (1, 2)$. The ellipse is $\mathcal{E}_\alpha = \{x \in \mathbb{R}^2 : x_1^2 + 2x_2^2 = 1\}$. The paths $\lambda \in \mathbb{R}_{>0} \mapsto p_{\xi_i}(\lambda) \in \mathbb{R}^2$ are displayed in red for two distinct points $\xi_1, \xi_2 \in \mathcal{E}_\alpha$ on the ellipse. The outer normal unit vector of the ellipse $\mathcal{E}_\alpha$ at ξ_1 is denoted by $\vec{n}_{\xi_1}$. The tangential component of the gradient at ξ_1 of an $(\alpha, k)-$homogeneous f is denoted $\iota_{\xi_1}(\nabla_t f(\xi_1))$ and lies in the linear space spanned by the tangent line of the ellipse $\mathcal{E}_\alpha$ at ξ_1. 160

Chapter 1

Introduction

In recent years, deep neural networks have made significant advances in various fields of artificial intelligence (AI), such as image recognition, image generation, text generation, and reinforcement learning. These advances have revolutionized many industries, including self-driving cars, creative design, natural language processing, and game playing. Specifically, self-driving cars rely heavily on deep neural networks for image recognition, while DALL-E generates images from text descriptions, GPT-4 generates coherent text, and AlphaGo Zero uses reinforcement learning to become the strongest Go player in history. These breakthroughs demonstrate the power and potential of deep neural networks to solve complex problems and drive innovation in the field of AI.

Deep learning has achieved great success due to the abundance of data, powerful computing resources, and advanced optimization techniques. However, while the practical advancements in deep learning have been remarkable, the theoretical understanding of this field is still at its onset. We remain unable to fully explain the effectiveness of the relatively simple tools used by deep learning practitioners. The purpose of this thesis is to explore various important theoretical aspects of deep learning by examining how the system behaves as one of its parameters approaches its limit – an approach known as *asymptotic analysis.*

1.1 Learning theory framwework

A good starting point to analyse deep learning is through the lens of a supervised learning problem. Suppose that the data lives in the space $\mathcal{Z}$, and let $\mathcal{D} \in \mathcal{P}(Z)$ be the data distribution. We seek to understand $\mathcal{D}$ via the means of a class of predictors

$\mathcal{F}$ and a loss function $\ell\colon \mathcal{F} \times \mathcal{Z} \to \mathbb{R}_+$. [1] Our goal is to minimize the *population* risk $R(f) \coloneqq \mathbb{E}_{z \sim \mathcal{D}}\left[\ell(f, z)\right]$.

In practice, we only have access to a set of samples S drawn from $\mathcal{D}$, which we assume to be independent and identically distributed (i.i.d.). It is not a realistic assumption, since $\mathcal{D}$ is usually an abstract concept, such as a uniform distribution of the set of "natural" images, and there is usually no good definition of independence between samples. Nevertheless, the i.i.d. assumption is convenient from a theoretical point of view. Define the *empirical* risk $R_S(f) \coloneqq \frac{1}{|S|} \sum_{z \in S} \ell(f, z)$.

Suppose that there exists an optimal solution that minimizes the (intractable) population risk: $f_{\text{opt}} \in \operatorname{argmin}_{f \in \mathcal{F}} R(f)$. Suppose as well that we have an algorithm $\mathcal{A}$ that takes a sample set S and outputs a predictor $\mathcal{A}_S \in \mathcal{F}$ based only on the sample set S. The population risk can be decomposed into the following error terms [23]:

$$R(\mathcal{A}_S) = \underbrace{R(\mathcal{A}_S) - R_S(\mathcal{A}_S)}_{generalization} + \underbrace{R_S(\mathcal{A}_S) - R_S(f_{\text{opt}})}_{optimization} + \underbrace{R_S(f_{\text{opt}}) - R(f_{\text{opt}})}_{concentration} + \underbrace{R(f_{\text{opt}})}_{approximation} \tag{1.1}$$

Note that while the concentration and the generalization terms look similar, they are handled in different ways. For the concentration term, as f_{opt} is independent of the sample set S, the random variables $\{\ell(f_{\text{opt}}, z_i)\colon i = 1, \ldots |S|\}$ are independent and identically distributed. Therefore, for $\delta > 0$, if we assume that $\max_{f,z} \ell(f, z) \leq M$, we have by Hoeffding's inequality that with probability $1 - \delta$,

$$R_S(f_{\text{opt}}) \leq R(f_{\text{opt}}) + M\sqrt{\frac{\log(1/\delta)}{2\,|S|}} \tag{1.2}$$

and the bound is sharp. However, the generalization term has to be estimated differently, as $\ell(\mathcal{A}_S, z_i)$ are no longer independent of each other.

We should choose the class of models $\mathcal{F}$ with the following trade-off in mind.

(i) $\mathcal{F}$ must be large enough to contain functions f that can reach low population risk $R(f)$. This can be achieved by using *a priori* knowledge about the data.

(ii) $\mathcal{F}$ must be chosen such that the algorithm $\mathcal{A}$ is efficient to compute $\mathcal{A}_S \approx \operatorname{argmin}_{f \in \mathcal{F}} R_S(f)$. We must also ensure that $\mathcal{A}_S$ is converging to f_{opt} as $|S| \to \infty$.

(iii) $\mathcal{F}$ must be small enough to ensure that the *generalization error* $R(\mathcal{A}_S) - R_S(\mathcal{A}_S)$ is small. Generally, this gap decreases to zero as $|S| \to \infty$, but can remain very large if $\mathcal{F}$ is big.

[1] For example, $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is the input space, $\mathcal{Y}$ is the output space, and $\ell(f, (x, y)) = d(f(x), y)$, where d is a distance function on $\mathcal{Y}$.

In the light of the above trade-off, we expand in more details on the three main sources of error: approximation, optimization, and generalization.

1.1.1 Approximation

The class of predictors $\mathcal{F} = \{f_\theta \colon \theta \in \Theta\}$ is determined by two components: the parametric specification f_θ of elements of $\mathcal{F}$, called the *neural network architecture*, and the space of values Θ that the parameters live in.

Feedforward neural networks. In the seminal paper [132], the author introduced the original and most well-known architecture: the *feedforward neural network*[2]. Fix $L \in \mathbb{N}^*$ the *depth* of the neural network, and let $\mathcal{X} = \mathbb{R}^{d_0}$ be the input space and $\mathcal{Y} = \mathbb{R}^{d_L}$ be the output space. Let $(d_1, \ldots, d_{L-1}) \in (\mathbb{N}^*)^{L-1}$ be the dimensions of the hidden states, also called *widths.* A feedforward neural network is a parametric map $f_{\theta^{(L)}} \colon \mathcal{X} \to \mathcal{Y}$ such that $f_{\theta^{(L)}}(x) = h_L^{(x)}$, where we define recursively

$$h_{k+1}^x = \varphi_k \left(h_k^x,\, \theta_k^{(L)} \right) \text{ for } k = 0, \ldots, L-1, \quad h_0^x = x. \tag{1.3}$$

Here, h_k^x is called the k^{th} *hidden state*, and $\varphi_k \colon \mathbb{R}^{d_k} \times \Theta_k \to \mathbb{R}^{d_{k+1}}$ the k^{th} *layer mapping.* The mapping φ_k can take various forms, the most common building block being the composition of an affine map and an *activation function* $\rho \colon \mathbb{R} \to \mathbb{R}$, i.e.

$$\varphi_k\big(h,\, (W_k, b_k)\big) = \rho_{d_{k+1}}\big(W_k h + b_k\big), \tag{1.4}$$

where $\rho_d \colon \mathbb{R}^d \to \mathbb{R}^d$ is defined component-wise by $\rho_d(x)_i = \rho(x_i)$. The parameter $W_k \in \mathbb{R}^{d_{k+1} \times d_k}$ is called the *weight matrix* of layer k, the parameter $b_k \in \mathbb{R}^{d_{k+1}}$ is called the *bias vector*, and the layer mapping (1.4) is called *fully-connected*, as each coordinate of h_k contributes to each coordinate of h_{k+1}. Popular choices for the activation functions are the rectified linear unit (ReLU): $\rho(x) = \max(x, 0)$, the hyperbolic tangent $\rho = \tanh$, or the leaky ReLU $\rho(x) = \max(x, 0) + \alpha \min(x, 0)$.

Weight space. There are three main lines of work to define the space in which the parameters can take values in, each focusing on making one of the terms in (1.1) as small as possible.

(i) **Minimize approximation term.** If our goal is to solely minimize the approximation term, we should consider the full Euclidean space in which parameters take values in. In particular, we can apply Stone-Weierstrass [53] to $\mathcal{F}$ and prove

[2] also called multilayer perceptron

universal approximation of continuous functions on a compact set. Similar ideas lie behind the original universal approximation theorem for neural networks with 1 hidden layer[3], see [38]. However, the worse case error rate for these shallow networks is unattractive [115]: for instance, we need a neural network of width $\Omega(\varepsilon^{-d/r})$ to approximate d–dimensional C^r functions with accuracy ε. This adversarial phenomenon is known as the *curse of dimensionality*.

(ii) **Minimize optimization term.** If we restrict the weights to only take values that can be reached by the optimization algorithm, we could control the optimization error closely. This strategy is also appealing from a generalization perspective, since we have tools such as *stability* [24] or *implicit regularization* [119] to understand how the optimization algorithm affects the error on unseen data, see Section 1.1.3 for more details. We use the former method in Section 5.4 and the latter in Section 4.3. Unfortunately, both methods rely on a precise understanding of how the optimization algorithm works, which is only known in restricted settings.

(iii) **Minimize generalization term.** To obtain good generalization properties, one can enforce *explicit regularization* on the parameter values. A popular technique is to consider $\Theta = \{\theta\colon \|\theta - \theta_0\| < R\}$, where θ_0 is the initial parameter value and $R > 0$. As we will see in Section 1.2.1, networks with small R and large width[4] are almost linear, and they satisfy a universal approximation theorem via RKHS theory. These results were established in the seminal paper on the *neural tangent kernel* (NTK) regime [82]. However, the width necessary to enter into the NTK regime is prohibitively large, and, empirically, networks that generalize well are usually far away from their initialization [10, 122].

Representational benefits of depth. The advantages of increasing the depth in lieu of the width has been well-documented for fully connected networks with ReLU activation. In [147], the author constructs a classification problem for which a network with depth L and width $O(1)$ can perfectly fit the training set, but where every network of depth $\mathcal{O}(1)$ and width $\mathcal{O}(\exp(L))$ yields a training error greater than 1/6. Other constructions and worse case error bounds are given in [159, 100], supporting the evidence that shallow networks require exponentially more width than deep networks to reach the same error level. The intuition behind this counter-example

[3] i.e. depth $L = 2$

[4] Models with large width are also called *overparametrized by width*

is that depth performs function compositions, which multiplies the number of linear pieces, whereas width perform function additions, which adds up the number of linear pieces. However, the examples built are highly pathological and unlikely to appear in a useful supervised learning problem.

Architecture modifications Modifications to the above architecture have been suggested mostly for two reasons: to decrease the generalization error or to speed up and/or stabilize the training. For example, if we incorporate local connectivity and weight sharing in A_k, we obtain a *convolutional* layer, widely successful in image and speech recognition [94]. Further, *batch/layer* normalization [79] were introduced to normalize the distribution of hidden states, allowing to train deeper networks. Also, the activation function can also be multivariate, such as a softmax layer for the last block of a neural network used for classification, or an attention layer [149] to emphasize one part of the hidden state over another. We can also introduce *skip connections* to design a *residual network* architecture, which we will introduce more precisely in Section 1.2.

1.1.2 Optimization

Recall from (1.1) that the optimization problem focuses on finding an algorithm $\mathcal{A}$ that minimizes the empirical risk $R_S(\mathcal{A}_S)$. However, the true objective that we are concerned about is the population risk $R(\mathcal{A}_S)$, and by the central limit theorem, $|R(\mathcal{A}_S) - R_S(\mathcal{A}_S)| \gtrsim |S|^{-1/2}$. Therefore, there is no apparent benefit to optimize the empirical risk at a scale smaller than $|S|^{-1/2}$, which sets apart the task of learning from pure optimization problems [60].

Stochastic gradient descent. A key ingredient in the success of deep learning is the ability to perform automatic differentiation to compute exact gradients of the loss function $R_S(f_\theta)$ with respect to θ. The process is called *backpropagation*, and there exist efficient implementations of backpropagation that require constant time and linear extra memory. As a result, first order methods such as stochastic gradient descent (SGD) have been the backbone of successful applications of deep learning. For an initial parameter value θ_0, the update rule of SGD reads, for $t \geq 0$:

$$\theta_{t+1} = \theta_t - \eta_t \nabla_\theta \ell(f_{\theta_t}, z_t), \quad z_t \overset{iid}{\sim} S \tag{1.5}$$

where $\eta_t > 0$ is called the *learning rate* at iteration time t. Even though the update rule (1.5) might not be a descent direction for R_S, θ_t converges (linearly) almost surely to a minimizer of R_S when $\theta \mapsto \ell(f_\theta, z)$ is strongly convex [23, 22]. Furthermore, there are theoretical and practical reasons to prefer SGD to (full batch) gradient descent:

(i) Typically, SGD achieves ε–optimality in $\mathcal{O}(1/\varepsilon)$ iterations, so $\mathcal{O}(1/\varepsilon)$ gradient evaluations, independently of the number of samples, whereas full-batch gradient descent achieves ε–optimality in $\mathcal{O}(\log(1/\varepsilon))$ iterations, hence $\mathcal{O}(|S|\log(1/\varepsilon))$ gradient evaluations [23]. Therefore, in the large data regime $|S| \gg 1$, one should always favor a stochastic gradient approach.

(ii) In practice, one observes a fast decrease in the error in the first few iterations of (1.5), hinting that at least for small t, the "signal" part $\nabla_\theta R_S(\theta_t)$ of the update dominates the "noise" part $\nabla_\theta \ell(f_{\theta_t}, z_t) - \nabla_\theta R_S(\theta_t)$. We will introduce in Section 1.2.3 a precise framework to understand both parts.

(iii) There exists methods to reduce the variance of the noise part. A widespread idea is to reuse previous gradient computations to get faster convergence rates, known as *gradient aggregation* [86, 40]. Another idea, not yet widely used in practice but theoretically sound, is *dynamic sample size methods*: instead of using one sample per iteration, one can utilize a dynamic number of samples at each iteration [23, Theorem 4.6]. We get the same complexity as the one-sample update (1.5), but with better bounds on the variance of the iterates.

Initialization. From an optimization perspective, the main challenge for initializing deep feedforward networks is avoiding vanishing or exploding gradients [76]. Via a heuristic argument aiming to preserve the variance of the forward and the backward states, [59] derived that the weights of a $m \times n$ linear layer with a sigmoid activation should be initialized with mean 0 and variance $2/(m+n)$, known as the *Xavier initialization*. [71] adapted the above derivation in the case of a ReLU activation to get a mean 0 and variance $2/m$ recommendation, known as the *He initialization*. In practice, both He and Xavier initializations are widely used, either with Gaussian or uniform weights, with no clear practical difference between the two distributions.
For deeper models, [129] performed a rigorous derivation of the average norm of the hidden states at initialization, when the weights, resp. biases, are centered Gaussian with variance σ_W^2, resp. σ_b^2. They obtained a phase transition in the (σ_W, σ_b) plane, separating it between two phases:

- Chaotic: Exploding gradients, expressivity of the network is retained, nearby input points can be separated.

- Ordered: Vanishing gradients, expressivity of the network is lost, nearby input points can not be separated anymore.

The trainable phase is therefore at the transition, called the *edge of chaos*. [125] computed the edge of chaos for all practical activation functions. Our work investigate the initialization strategies for residual networks, see Section 3.3 for numerical experiments and Section 4.3 for a theoretical investigation.

Loss landscape. Recall that the loss function we seek to minimize is $\mathcal{L}(\theta) \coloneqq R_S(f_\theta) = |S|^{-1} \sum_{z \in S} \ell(f_\theta, z)$. If the predictor f_θ is linear in θ, and the loss function ℓ is convex in its first variable, the empirical risk $\mathcal{L}$ is convex. As a result, there are no saddle points, and every local minimum is global. These properties ensure that first order methods are converging almost surely to a global minimum of $\mathcal{L}$. However, if f_θ is a neural network, then $\mathcal{L}$ is non-convex, and the sublevel set $\{\theta \colon \mathcal{L}(\theta) \leq \lambda\}$ has more than one single connected component, for some $\lambda > 0$ [55]. But with enough overparametrization by width, all sublevel sets are connected [55], and all local minima are global [120]. Note that in the linear case, the latter is true for any width [87]. Regarding saddle points, early empirical studies have shown that the loss landscape of neural network contained many saddle points [40], but they are almost surely not visited by gradient descent [96].

The above behaviour is reminiscent of many classes of random functions: the probability of a particular eigenvalue of the Hessian at a critical point being negative is around 1/2 in high-cost region of the loss function, and decreases as the cost decreases [26]. This means, when the number of parameters is high, there are exponentially more saddle points than local minima, and local minima are more likely to be found in low-cost regions.

1.1.3 Generalization

Recall that the generalization error in (1.1) is given by

$$\varepsilon_{\text{opt}}(S) \coloneqq R(\mathcal{A}_S) - R_S(\mathcal{A}_S) \leq \sup_{f \in \mathcal{F}} |R(f) - R_S(f)| \eqqcolon \varepsilon_{\text{opt}}. \tag{1.6}$$

Note first that using (1.2) together with a union bound yields a vacuous upper bound, as $|\mathcal{F}|$ is at least super-exponential in the number of parameters. One needs to resort to the Vapnik-Chervonenkis (VC) dimension of $\mathcal{F}$ to get a first non-trivial result:

$$\varepsilon_{\text{opt}} \lesssim C \sqrt{\frac{\text{VCdim}(\mathcal{F}) + \log(1/\delta)}{2\,|S|}}$$

$\text{VCdim}(\mathcal{F}) = \mathcal{O}(L)$, where $L_{\mathcal{F}}$ is the maximum depth of networks in $\mathcal{F}$, and $P_{\mathcal{F}}$ is the maximum number of parameters of networks in $\mathcal{F}$. However, as argued in multiple empirical studies [119, 162, 54], the number of parameters is not a good measure for the model capacity of neural networks. Also, studying the uniform bound ε_{opt} instead of $\varepsilon_{\text{opt}}(S)$ removes the dependence on the optimization algorithm, which seems to explain some of the out-of-sample success of neural networks [119, 162]. Therefore, through three puzzling empirical observations, we present modern approaches to understand the generalization ability of neural networks.

1. Explicit regularization is not the full story. The goal of regularization is to reduce the complexity[5] of the hypothesis space $\mathcal{F}$ in order to improve generalization. Several techniques have been applied successfully in the context of neural networks.[6]

(i) *Data augmentation:* Constructing new data points by perturbing the existing ones adversarially [62], via domain-specific transformation, or by using generative methods like variational autoencoders [90], generative adversarial networks, [61] or diffusion models [144].

(ii) *Sparsity/parameter sharing:* Altering the structure of the neural network to fit domain-specific applications. For example, convolutional layers share parameters across dimensions and the convolutional operator is sparse, which makes it robust to study local structures of the input such as images.

(iii) *Dropout:* Masking neural connections randomly with a given probability during training, and using the full network structure for the inference [145].

(iv) *Weight decay:* Scaling down the contribution of the parameters of the previous iteration in the update rule (1.5) of SGD. Weight decay is equivalent to adding a ℓ^2-penalty on the weights to the loss function.

(v) *Gradient penalty:* Adding a ℓ^2-penalty on the gradients of the loss function with respect to the weights. We will see in Section 1.2.3 following an analysis in [14] that this penalty arises naturally in the limit of the training step-size going to zero.

The relative success of explicit regularization inspired norm-based approaches to derive theoretical bounds, for example using Rademacher complexity and spectral norms

[5] also called *effective capacity*

[6] The list is not exhaustive, we present the most common ones.

[17], PAC-Bayes and noise robustness [11], or in a more direct way using path norms [88]. However, even with all explicit regularizations turned off, the models can still memorize the training set and show no sign of overfitting [92, 162].

2. The optimization algorithm affects generalization. The effect of the optimization algorithm on the generalization properties of the neural network at convergence is often called *implicit regularization*. To understand what it entails, we consider a simple linear regression problem [64]: we have the dataset $S = \{(x_i, y_i) \colon i = 1, \ldots, n\}$, where x_i are d-dimensional feature vectors, and y_i is their corresponding 1-dimensional target. Let $X \in \mathbb{R}^{n \times d}$ such that the i^{th} row of X is x_i, and assume that $d > n$, X has full rank, and ℓ is convex in its first argument. The empirical risk of a weight vector $\beta \in \mathbb{R}^d$ is thus given by

$$R_S(\beta) = \frac{1}{n} \sum_{i=1}^{n} \ell(\beta^\top x_i, y_i)$$

Observe that R_S is convex, so SGD will converge to a global minimum. As $d > n$ and X has rank n, there exist infinitely many global minima, that is, weights β such that $R_S(\beta) = 0$. Which solution(s) will be reached by SGD? Assume $\beta_0 = 0$ and the SGD update rule (1.5) is given by

$$\beta_{t+1} = \beta_t - \eta_t \frac{\partial \ell}{\partial y_1} \left(\beta_t^\top x_{i_t}, y_{i_t}\right) x_{i_t}$$

for some $i_t \in \{1, \ldots, n\}$ chosen at random. Therefore, after a (possibly infinite) training time T, $\beta_T \in \operatorname{span}\{x_1, \ldots, x_n\}$, i.e. there exists $\alpha \in \mathbb{R}^n$ such that $\beta_T = X^\top \alpha$. Furthermore, as β_T is a global minimum, $X\beta_T = y$. Therefore, $XX^\top \alpha = y$, and as X has rank n, $XX^\top$ is invertible and $\beta_T = X^\top \left(XX^\top\right)^{-1} y$. One can verify that β solves the following minimization problem

$$\min \|\beta\|_2 \quad \text{such that } X\beta = y \tag{1.7}$$

In practice, β_T generalizes well, and one can construct other global minima (with higher norm) that generalize poorly. However, the minimum norm solution 1.7 is not a perfect predictor of generalization performance [162]. Similarly to linear regression, gradient descent for linear neural networks converges to the maximum margin solution [83], as well as for homogeneous 2-layer wide neural networks [32]. We extend this theory for deep neural networks with a smooth activation function in Section 4.3, and show that gradient descent converges to a global minimum that is regular as a function of the layer.

Another type of implicit regularization is the use of *early stopping* in the optimization algorithm. It is based on the observation that the risk on a validation set usually follows a U-shape curve as a function of the training time [60]. In this regime, training the network to global convergence hurts the population risk, and the number of training iterations T can be viewed as a hyperparameter to tune. In the context of linear models, [60] showed that early stopping is equivalent to ℓ^2-regularization with coefficient inversely proportional to $\sum_{t=0}^{T} \eta_t$.

A vast literature explores how to derive generalization bounds for optimization algorithms, for example algorithmic robustness [155], uniform stability [24, 69] or information-theoretic stability [2, 136, 154], among others. We derive uniform stability bounds for a class of homogeneous models in the mean-field limit optimized with gradient flow in Section 5.4. The assumptions required to verify those bounds are usually weak, at the expense of ignoring the underlying data distribution or the neural network architecture. This can lead to vacuous bounds under input or label noise [162], and uniform convergence is not able to fully explain generalization [116].

3. Overparametrization helps generalization Traditional statistics rely on the bias-variance trade-off to choose the optimal complexity of a model: the out-of-sample error is usually a U-shape curve as a function of the model complexity - usually related to the number of free parameters in parametric statistics. As the complexity increases, the model is able to fit the training samples and reduce the bias, at the expense of a higher variance until the *interpolation threshold* is reached, above which there is no bias and an exploding variance.

Empirically, neural networks operate optimally well beyond the interpolation threshold, when the number of parameters d far exceeds the number of samples n. The data can be perfectly interpolated, even when the targets are corrupted with noise [162], and the out-of-sample error is lower than in the underparametrized regime. This phenomenon is called *double descent* in the literature.

Even though double descent was discovered in the context of deep learning, it is not idiosyncratic to neural networks and can be observed in linear regression with random Fourier feature and tree-based algorithm [19]. Rigorous proofs of this phenomenon exist for linear regression with random covariates [18], and random features regression (related to kernel methods) [113]. To give an intuition on the former, consider the case $n = 1$ and $\mathcal{X} = \{-1, 1\}^d$. After normalization, the population risk of the minimum norm interpolator[7] β^* in (1.7) is $R(\beta^*) = d^{-2} \left\| \mathbb{E}_{X \sim \mathcal{D}} \left[X X^\top \right] \right\|_F^2$, see [18]. Therefore,

[7] Equivalently, the large-time limit of the gradient flow with zero initialization.

in the overparametrized regime $d > 1$, increasing the feature dimension from d to $d+1$ can increase or decrease the population risk depending on the correlation of the new feature X_{d+1} with respect to the other features $X_{1:d}$.

It is worth mentioning that double descent is closely related to the implicit regularization of gradient descent methods, as it is easy to engineer overparametrized neural networks that interpolate the training data, but have a poor out-of-sample performance.

1.2 Asymptotics

Asymptotic analysis plays a fundamental role in science by providing a powerful tool to understand the behavior of systems as certain parameters tend to infinity. This analysis is particularly useful in the study of complex systems, where direct analytical solutions are not available, to gain insight into their long-term/large scale properties and identify their dominant behaviors. For example, asymptotic analysis is central in the complexity analysis as a function of the input size in computer science, the performance of high-pass and low-pass filters as a function of the frequency in electrical engineering, and the behavior of a viscous fluid flow as a function of the Reynolds numbers.

We look at three different limits of the deep learning framework: the width and the depth of the neural network going to infinity, and the learning rate of SGD going to zero. Other limits can be investigated, such as when the number of data points or the computational budget tend to infinity, but they lie beyond the score of this thesis.

1.2.1 Width goes to infinity

We revisit the original feedforward architecture described in (1.3) and incorporate a scaling factor in front of the weight matrix that depends on the width:

$$h^x_{k+1} = d_k^{-1/2} W_k \, \rho_{d_k}\left(h^x_k\right) + b_k \quad \text{for } k = 1, \dots, L-1 \quad \text{and} \quad h^x_1 = d_0^{-1/2} W_0 \, x + b_0,$$

where $W_k \in \mathbb{R}^{d_{k+1} \times d_k}$ and $b_k \in \mathbb{R}^{d_{k+1}}$ are the weight matrix and the bias vector of the k^{th} layer, and $\theta = \{(W_k, b_k) \colon k = 0, \dots, L-1\}$ is the full parameter set. Assume that the output is 1–dimensional, i.e. $d_L = 1$, and define $f_\theta(x) = h^x_L$ the output of the last layer. The scaling is consistent with the popular initialization schemes introduced in Section 1.1.2, and the $d_k^{-1/2}$ factor is crucial to get a consistent limiting behaviour as the widths $d_1, \dots, d_{L-1}$ tend to infinity.

We now informally derive the exact dynamics of the neural network f along the gradient flow of the parameters in the limit $d_1, \dots, d_{L-1} \to \infty$ using kernel regression. This was first observed in [35] for $L = 2$, and rigorously proven in a general setting in [82], which called the underlying object the *neural tangent kernel*.

Neural tangent kernel Assume that θ_0 is initialized with independent centred Gaussian entries, with variance σ_W^2 for the weights and σ_b^2 for the biases. To simplify the analysis, assume that the parameters are following the gradient flow of the empirical risk on the dataset $S = \{(x_i, y_i) \colon i = 1, \dots, n\}$.

$$\frac{\mathrm{d}\theta_t}{\mathrm{d}t} = -\nabla_\theta R_S\left(f_\theta\right)\big|_{\theta=\theta_t} = -\frac{1}{n}\sum_{i=1}^{n} \nabla_\theta f_{\theta_t}(x_i) e_t(x_i, y_i), \quad e_t(x,y) = \frac{\partial \ell}{\partial y_1}(f_{\theta_t}(x), y). \tag{1.8}$$

We deduce the following dynamics for the neural network realization and the empirical risk.

$$\frac{\partial f_{\theta_t}}{\partial t} = -\frac{1}{n}\sum_{i=1}^{n} K_{\theta_t}(\,\cdot\,, x_i) e_t(x_i, y_i) \tag{1.9}$$

$$\frac{\partial}{\partial t} R_S\left(f_{\theta_t}\right) = -\frac{1}{n^2}\sum_{i,j=1}^{n} e_t(x_i, y_i) K_{\theta_t}(x_i, x_j) e_t(x_j, y_j) \tag{1.10}$$

where the neural tangent kernel (NTK) $K_\theta \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is given by

$$K_\theta(x, x') = \nabla_\theta f_\theta(x)^\top \nabla_\theta f_\theta(x'). \tag{1.11}$$

Under weak assumptions on the activation function ρ, the hidden states h_k^x converge as $d_1, \dots, d_{L-1} \to \infty$ to i.i.d. centered Gaussian process with variance defined by a recurrence equation [111]. Also, K_{θ_0} converges to a deterministic kernel K^∞ which is constant in t and only depends on ρ, the depth L, and the initial variance σ_W and σ_b [82, 122]. Further, in this limit, as long as the kernel matrix $\overline{K}^\infty := (K^\infty(x_i, x_j))_{i,j=1}^m$ is positive definite, $R_S\left(f_{\theta_t}\right)$ converges to zero. When ℓ is the quadratic loss $\ell(y_1, y_2) = |y_1 - y_2|^2$, the solution of (1.9) can be found explicitly:

$$f_{\theta_t} = D_t(\,\cdot\,)Y + \left[f_{\theta_0} - D_t(\,\cdot\,) f_{\theta_0}(X)\right] \tag{1.12}$$

where $Y = (y_1, \dots, y_n)^\top \in \mathbb{R}^n$, $f_{\theta_0}(X) = (f_{\theta_0}(x_1), \dots, f_{\theta_0}(x_n))^\top \in \mathbb{R}^n$, and

$$D_t \colon \mathcal{X} \to \mathbb{R}^n, \quad D_t(x) = \left(K^\infty(x, x_i)_{i=1}^m\right)^\top \left(\overline{K}^\infty\right)^{-1} \left(I_n - \exp\left(-\frac{2t}{m}\overline{K}^\infty\right)\right).$$

We directly see from kernel regression theory that the first term in (1.12) converges as $t \to \infty$ to the minimum RKHS-norm interpolator:

$$\min_{f \in \mathcal{H}_{K^\infty}} \|f\|_{K^\infty} \quad \text{s.t.} \quad f(x_i) = y_i.$$

The second term in 1.12 is a Gaussian process with zero mean and variance vanishing on the data points x_i, $i = 1, \ldots, n$, because $D_t(x_i) \to e_i$ as $t \to \infty$. Finally, we have formally in the limit $d_1, \ldots, d_{L-1} \to \infty$ that

$$f_{\theta_t} \approx f_{\theta_t}^{\text{lin}} := f_{\theta_0} + (\theta_t - \theta_0)^\top \nabla_\theta f_{\theta_0}.$$

The approximation and convergence rates can be made precise [95]. Essentially, in the infinite-width limit, the neural network behaves like a linear model on the (non-linear) features $\nabla_\theta f_{\theta_0}(x)$. This is called *lazy training*, and it can already be observed in a non-asymptotic regime.

Lazy training The lazy regime arises naturally as a consequence of overparametrization, or at the beginning of the optimization. Indeed, for the 2–layer case, if the activation function ρ is β-smooth, then Taylor's theorem directly gives

$$\left| f_{\theta_t}(x) - f_{\theta_t}^{\text{lin}}(x) \right| \leq \frac{\beta}{2\sqrt{d_1}} \|\theta_t - \theta_0\|_F^2 .$$

Several papers have utilized this observation to establish the global convergence of (stochastic) gradient descent for neural networks overparametrized by width. For example, [45] proves that for the 2-layer ReLU case with $d_1 = \Omega(n^6 \delta^{-3})$, we have linear convergence of the empirical risk to zero with probability $1 - \delta$ over the random initialization. [33] considers a more general setting, where a simple final layer scaling can put any model into the lazy regime, and also provides linear convergence rates. However, [10] reports that empirically, networks trained in the lazy regime lag behind their finite-width counterparts in terms of out-of-sample accuracy. Therefore, despite encouraging theoretical bounds on the generalization gap in the lazy regime [27], overparametrization alone cannot explain the ability of neural networks to generalize well.

In Section 4.3, we study global convergence of the empirical risk for deep residual networks with smooth activation. Our analysis goes beyond the lazy regime, and even if the parameters found by gradient descent are far from their initialization, we can still guarantee convergence thanks to the implicit regularization displayed by gradient descent.

Mean-field limit A crucial assumption in the NTK regime is the $d_k^{-1/2}$ scaling of the weights, leading to a linear evolution of the realization function f_{θ_t} under the gradient flow of the empirical risk and the parameters staying close to their initialization during training: $\sup_t \|(W_k)_t - (W_k)_0\|_2 = \mathcal{O}(d_k^{-1/2})$. However, in practice, we already have for small t: $\|\theta_t - \theta_0\|_2 = \mathcal{O}(1)$. This observation motivated the study of another scaling: for the two-layer neural network:

$$f_\theta(x) = \frac{1}{d} \sum_{i=1}^{d} c_i \rho(w_i^\top x), \text{ where } \theta = (w, c) \in \mathbb{R}^{d \times d_0} \times \mathbb{R}^d. \tag{1.13}$$

Let $\mu_d \in \mathcal{M}(\mathbb{R}^{d_0})$ be the (signed) measure of the parameters: $\mu_d = d^{-1} \sum_{i=1}^d c_i \delta_{w_i}$. Then

$$f_\theta(x) = \int_{\mathbb{R}^{d_0}} \rho(w^\top x) \mathrm{d}\mu_d(w) = \left(\int_{\mathbb{R}^{d_0}} \Phi(w) \mathrm{d}\mu_d(w) \right)(x),$$

where $\Phi \colon \mathbb{R}^{d_0} \to \mathcal{F}$ is defined by $\Phi(w) = \rho(\langle w, \cdot \rangle)$. The evolution of μ_d under the gradient flow of the empirical risk R_S is thus given by

$$\partial_t \, \mu_{d,t} = -\mathrm{div} \left(\left\langle R_S' \left(\int \Phi \mathrm{d}\mu_{d,t} \right), \nabla \Phi \right\rangle_{\mathcal{F}^*} \mu_{d,t} \right), \tag{1.14}$$

where R_S' is the Frechet derivative of R_S. The infinite-width limit in (1.14) is obtained by letting $d \to \infty$, which yields a non-linear PDE in the space of measures. Concurrent papers have established the link between the two-layer neural network (1.13) and its infinite-width limit during training: [133] from the interactive particles and fluctuations perspective and [141] from a mean-field probabilistic approach. Under strong assumptions on the data distribution, [114] established convergence as $t \to \infty$ of noisy SGD to a global minimum for d big enough. Under a homogeneity condition on the model, [31] studied the training dynamics in the limit $d \to \infty$ in the Wasserstein space of probability measures. They also derived convergence to a global minimum as $t \to \infty$.

Contrary to the NTK regime, the mean-field dynamics of the neural network realization are non-linear, so the network can potentially represent a larger class of functions. However, the mean-field theory does not extend naturally for more than one hidden layer. Potential solutions have been proposed, such as taking the successive limits $d_{L-1} \to \infty$, and so on until $d_1 \to \infty$ [142], or considering the connected paths from an input neuron to an output neuron as the building block for the mean-field construction [6]. Furthermore, little is known about the convergence rate to global minima and generalization error bounds in the mean-field regime.

Contributions In Chapter 5, we study the mean-field limit of a particular class of models called *path-homogeneous*, which encompasses deep neural networks with special local connectivity. We show convergence to a global minimum, and we prove a generalization bound based on the algorithmic stability of the optimization method used: the Wasserstein gradient flow.

1.2.2 Depth goes to infinity

As mentioned above, the problem of vanishing/exploding gradient [20] in the training and in the performance of **deep** feedforward neural networks has restricted the use of very large depth. *Ad hoc* solutions have been introduced to fix this issue, such as careful initialization [59, 71] with batch normalization layers [79] to allow the gradients to stay within the same magnitude throughout the network. Yet, the existing literature rarely reports successful training and good generalization results for networks with more than 100 layers. In contrast, adding a skip connection at every layer, resulting in a *residual network*, allows the successful training of networks with thousands of layers, together with a better generalization performance than the best feedforward networks [73]. For example, ResNet-1001 [72], which consists of 1000 layers, achieves a 4.92% out-of-sample error on the CIFAR-10 dataset.

Residual networks There are multiple different residual architectures used in practice, each of them fine-tuned to a particular problem. Instead of listing them exhaustively, we give a general form of the ResNet as described in [72]. Let $\mathcal{X} = \mathbb{R}^{d_0}$ be the input space and $\mathcal{Y} = \mathbb{R}^{d_L}$ be the output space. Fix $L \in \mathbb{N}^*$, and $(d_1, \ldots, d_L) \in (\mathbb{N}^*)^L$ the dimensions of the hidden states. Let $x \in \mathcal{X}$ be an input vector, and $\psi^{d,d'} : \mathbb{R}^d \to \mathbb{R}^{d'}$ be either the projection onto the first d' components if $d \geq d'$, or the padding with zeros on the last $d' - d$ components if $d < d'$. Define a residual network as the parametric input-output map $x \in \mathcal{X} \mapsto f_{\theta^{(L)}}(x) \in \mathcal{Y}$ given by

$$\begin{cases} h_0^{x,(L)} &= g_{\text{in}}\left(x, \theta_0^{(L)}\right), \\ h_k^{x,(L)} &= \psi^{d_{k-1},d_k}\left(h_{k-1}^{x,(L)}\right) + \mathcal{F}_k\left(h_{k-1}^{x,(L)},\, \theta_k^{(L)}\right) \quad \text{for } k = 1, \ldots, L, \\ f_{\theta^{(L)}}(x) &= g_{\text{out}}\left(h_L^{x,(L)}, \theta_{L+1}^{(L)}\right). \end{cases} \tag{1.15}$$

Here, we let $\mathcal{F}_k \colon \mathbb{R}^{d_{k-1}} \times \Theta_k^{(L)} \to \mathbb{R}^{d_k}$ be a *residual block*, composed usually of a composition of linear layers and non-linear activations. The term $\psi^{d_{k-1},d_k}\left(h_{k-1}^{x,(L)}\right)$ is called a *skip connection*. The input downsampling is given by the parametric function

$g_{\text{in}} \colon \mathcal{X} \times \Theta_0^{(L)} \to \mathbb{R}^{d_0}$ and the prediction layer is given by the parametric function $g_{\text{out}} \colon \mathbb{R}^{d_L} \times \Theta_{L+1}^{(L)} \to \mathcal{Y}$. The set of parameters of the residual network is thus

$$\theta^{(L)} := \left\{ \theta_k^{(L)} \colon k = 0, \dots, L+1 \right\}.$$

The general form (1.15) is difficult to analyse systematically, so we assume in the following that $d_0 = \cdots = d_L = d$ and g_{in} and g_{out} are identity functions in the first argument. The update rule now reads

$$h_k^{x,(L)} = h_{k-1}^{x,(L)} + \mathcal{F}_k \left(h_{k-1}^{x,(L)}, \theta_k^{(L)} \right) \text{ for } k = 1, \dots, L \quad \text{and} \quad h_0^{x,(L)} = x. \tag{1.16}$$

Neural ODEs A series of papers have observed and utilized empirically the perceived link between the forward dynamics of residual networks (1.16) and the discretization of ordinary differential equations. For example, [67] develop new architectures inspired by the stability condition of the Euler scheme. In the same vein, [107] build on the so-called *linear multi-step* scheme in numerical ODE to construct a novel deep architecture. These insights culminated with the *neural ODE* architecture [28], which uses adaptive step-size schemes to adjust the depth needed to the function it is trying to fit. The network parameters are shared across all layers, making this approach appealing in terms of memory.

However, the link with ordinary differential equations is only formal. A first issue, often overlooked, is that there is no explicit scaling factor in front of the activation in (1.16). Indeed, explicitly scaling the residual network with $1/L$ empirically hurts performance [13]. Also, there is no guarantee that the parameters learnt by the optimization algorithm will have a scaling limit when $L \to \infty$, as you would expect from discretization schemes of ODEs.

Contributions In Chapter 2, we study linear residual networks as a toy model for non-linear networks, and describe explicitly all the minima of the empirical risk. We then show global convergence of the gradient flow to a minimum which admits a scaling limit. This property demonstrates the implicit regularization of the gradient flow for deep linear residual network.

In Chapter 3, we investigate the asymptotic properties of deep residual networks as the number of layers increases. We first show the existence of scaling regimes for trained weights markedly different from those implicitly assumed in the neural ODE literature. We study the convergence of the hidden state dynamics in these scaling regimes, showing that one may obtain an ODE, a stochastic differential equation (SDE), or neither. In particular, our findings point to the existence of a diffusive regime in which the deep network limit is described by a class of stochastic differential equations (SDEs). Finally, we derive the corresponding scaling limits for the backpropagation dynamics.

In Chapter 4, we prove local linear convergence of gradient descent to a global minimum for the training of deep residual networks with constant layer width and smooth activation function. We show that if the trained weights, as a function of the layer index, admit a scaling limit as the depth increases, then the limit has finite quadratic variation.

1.2.3 Learning rate goes to zero

Recall the stochastic gradient descent update rule (1.5) with constant learning rate $\eta > 0$:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \ell(f_{\theta_t}, z_t), \quad z_t \overset{iid}{\sim} \mathrm{Unif}(S)$$

for optimizing the loss function $L(\theta) = R_S(f_\theta) = \frac{1}{|S|}\sum_{z\in S} \ell(f_\theta, z)$. First, we show that the discrete SGD trajectory is close (in expectation) to the gradient flow of a regularized objective when η is small [14, 143].

Implicit regularization of the discretization Consider first the (full-batch) gradient descent update $\theta_{t+1} = \theta_t - \eta\nabla_\theta L(\theta)$. Suppose that $\bar\theta$ evolves according to the regularized gradient flow: $\partial_t \bar\theta_t = -\nabla_\theta \left(L(\bar\theta_t) + \eta N(\bar\theta_t)\right)$, for some $N : \Theta \to \mathbb{R}_+$. Then, a second-order expansion in η leads to

$$\begin{aligned}\bar\theta_{t+\eta} &= \bar\theta_t + \eta\partial_t\bar\theta_t + \frac{1}{2}\eta^2\partial_t^2\bar\theta_t + \mathcal{O}(\eta^3)\\ &= \bar\theta_t - \eta\nabla_\theta L(\bar\theta_t) + \eta^2\left(-\nabla_\theta N(\bar\theta_t) + \frac{1}{2}\nabla_\theta^2 L(\bar\theta_t)\nabla_\theta L(\bar\theta_t)\right) + \mathcal{O}(\eta^3).\end{aligned}$$

Therefore, if we let $N(\theta) := \frac{1}{4}\left\|\nabla_\theta L(\theta)\right\|^2$, Gronwall's lemma ensures that if L is β-smooth, then $\left\|\theta_k - \bar\theta_{k\eta}\right\| = \beta^{-1}\left(\exp(\beta k\eta) - 1\right)\mathcal{O}(\eta^2)$, as opposed to $\mathcal{O}(\eta)$ without the regularization term $N(\,\cdot\,)$. [143] extended the analysis for the (average) SGD update rule, which approximates the gradient flow of

$$L_{\mathrm{SGD}}(\theta) = L(\theta) + \frac{\eta}{4}\left\|\nabla_\theta L(\theta)\right\|^2 + \frac{\eta}{4\,|S|}\sum_{z\in S}\left\|\nabla_\theta \ell(f_\theta, z) - \nabla_\theta L(\theta)\right\|^2.$$

The third term penalizes the covariance of the stochastic updates, which motivates the below SDE (1.18) in the limit $\eta \to 0$.

SGD is the discretization of a SDE An important component of SGD that we omitted so far is the use of a (random) *mini-batch* of training data to perform the SGD update step. Namely, let $B \in \mathbb{N}$ be the *batch size*, and let $\Gamma \in \mathcal{X}^B$ be a uniform random variable on the subsets of S of size B. The SGD update rule is then

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta L_{\Gamma_t}(\theta_t), \tag{1.17}$$

where $\Gamma_t \overset{iid}{\sim} \Gamma$ and $L_\Gamma(\theta) \coloneqq \frac{1}{B}\sum_{z\in\Gamma} \ell(f_\theta, z)$. We split in (1.17) the signal and the noise part of the gradient:

$$\theta_{t+1} = \theta_t - \eta\nabla_\theta L(\theta_t) + \left(\eta\left(\frac{1}{B} - \frac{1}{|S|}\right)\Sigma(\theta_t)\right)^{1/2}\sqrt{\eta}Z_t,$$

where $\mathbb{E}_{\gamma\sim\Gamma}[Z_t] = 0$ and $\mathrm{Cov}_{\gamma\sim\Gamma}[Z_t] = I_d$, and where we assume that the covariance of the gradient noise is finite:

$$\Sigma(\theta) \coloneqq \frac{1}{|S|-1}\sum_{z\in S}\left(\nabla_\theta\ell(f_\theta, z) - \nabla_\theta L(\theta)\right)\left(\nabla_\theta\ell(f_\theta, z) - \nabla_\theta L(\theta)\right)^\top < \infty.$$

Under strong smoothness assumptions on the growth of L and its derivatives [97, Theorem 1], we can prove that for each $T \geq 0$, the SDE

$$\mathrm{d}\Theta_t = -\nabla_\theta L(\Theta_t) + \left(\eta\left(\frac{1}{B} - \frac{1}{|S|}\right)\Sigma(\Theta_t)\right)^{1/2}\mathrm{d}W_t, \quad (W_t)_{t\in[0,T]} \text{ is a BM}, \tag{1.18}$$

is an order-1 weak approximation of (1.17), that means, the difference of expectations of test functions applied to (1.17) and (1.18) is $\mathcal{O}(\eta)$. The approximation is appealing as [30] showed empirically that the third (and higher) moments of the gradient noise do not influence the convergence or the generalization ability of SGD. However, as $\eta \to 0$, the noise vanishes and the SGD path becomes closer to the gradient flow of L, whose solution is known to generalize poorly compared to its stochastic counterpart with small but non-vanishing learning rate [21, 89]. Furthermore, applied researchers observed that the *linear scaling rule*, which consists of keeping the *scale of the noise* η/B constant during training, leads to optimal out-of-sample performance [63, 108]. Since then, the scale of the noise appeared in a generalization bound [70] derived using algorithmic stability, and in an optimality criterion for SGD in the basin of attraction of a local minimum [110]. Therefore, it is not clear whether the continuous-time SDE (1.18) is a useful tool to understand the convergence and generalization properties of SGD. In response, [99] introduced practical conditions to check whether the approximation holds in practice.

Contributions In Section 2.4 and Chapter 5, we look directly at the gradient flow for linear residual networks and path homogeneous models. The connection between gradient flow and gradient descent is well understood, and we expect similar results for discrete updates. In Section 4.3, we prove a local convergence result for gradient descent for residual networks with a smooth activation. A crucial assumption is that the learning rates must be small, but their cumulative sum should still diverge to guarantee convergence.

Chapter 2

Linear residual networks

A good first stepping stone in the quest to understand general neural networks architectures is to first consider the case of a linear activation. For example, define the following parametric model: $g_{(W_1,W_2)}(x) = W_2 W_1 x$. It can only represent linear maps, but the landscape of the mean-square training loss function $f(W_1, W_2) = \mathbb{E}\left[(y - W_2 W_1 x)^2\right]$ is not convex in (W_1, W_2). Hence, this landscape possibly contains suboptimal critical points and local optima, which would hinder the power of first-order optimization methods like (stochastic) gradient descent. However, we know exactly what the product $W_2 W_1$ should be, as f is convex in $W_2 W_1$. There are efficient algorithms to solve convex problems, such as interior point methods or subgradient methods, with theoretical guarantees [25]. Hence, we can use this knowledge to characterize the set of minima, and to analyse which solution is picked by first-order methods such as gradient descent with a generic initialization.

In analogy with a residual neural network [73], we call the following parametric model a *linear residual network* with L layers

$$g_{A^{(L)}}(x) = \prod_{k=1}^{L} \left(I_d + A_k^{(L)}\right) \cdot x,$$

where $A^{(L)} = \left(A_k^{(L)}\right)_{k=1,\dots,L} \in \mathbb{R}^{L\times d\times d}$ are the parameters of the model, also called *weight matrices.*

2.1 Recent work

The problem of supervised training of residual linear networks has been well-studied over the last few years.

For the regression problem, [68] prove that around the origin, all critical points of the mean-squared loss function are global minima, and there exists a solution where each matrix $A_k^{(L)}$ is of the order $1/L$. More generally, [87, 93, 104] all establish that the mean-squared loss admits no suboptimal local minima, each time under slightly different assumptions.
Furthermore, [7] study the gradient descent dynamics starting from a *balanced initialization*, and show that the mean-squared error drops below ε in $\mathcal{O}\left(L^3 \log(1/\varepsilon)\right)$ iterations. Similarly, [152] argue that an asymmetric initialization of the weight matrices yields a convergence of gradient descent in $\mathcal{O}\left(L^3 \log(1/\varepsilon)\right)$ iterations as well, but with better constants. They also show that learning the linear map $x \mapsto -x$ with gradient descent initialized using the popular *Xavier initialization* [59] fails to converge to the global minimum in less than $\Omega(\exp(L))$ iterations.
Also, several works study the problem of training linear networks when the response Y does not necessarily come from noisy observations of an underlying linear model. For example, [58] study the properties of the solution found by gradient descent starting from a Xavier initialization in the case of $L = 1$ and $L = 2$ layers.

For the classification problem, [84] establish that training deep linear networks on linearly separable data with gradient descent on the logistic loss converges to a global minimum that aligns the weight matrices along the depth. In other words, the trained matrices are asymptotically of rank 1, and can be seen as a form of implicit regularization. [65] study the case of linear convolutional networks of depth L trained on linearly separable data with gradient descent on the exponential loss. They show that the rescaled input-output map converges to the hard-margin support vector machine classifier with minimal $\ell_{2/L}$ norm. Hence, gradient descent without explicit regularization is biased towards sparse solutions for $L = 2$.

2.2 Problem formulation

We take the setup from [68] and assume that the input distribution $\mathcal{D}$ lies in $\mathbb{R}^d$, and $R\colon \mathbb{R}^d \to \mathbb{R}^d$ is a linear mapping. Let X be a square-integrable random variable defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ such that X is distributed according to $\mathcal{D}$, and let $Z \sim \mathcal{N}(0, \xi I_d)$ be independent of X. Denote $\Sigma := \mathbb{E}\left[XX^T\right] \in \mathbb{R}^{d \times d}$ the covariance matrix of X.

We seek to learn the mapping R from noisy measurements $Y = RX + Z$. To do so, we use a linear residual network of depth $L \in \mathbb{N}^*$, that is, we parametrize our model with weights matrices $A^{(L)} = \left(A_1^{(L)} \dots, A_L^{(L)}\right) \in \mathbb{R}^{L \times d \times d}$ so that for an input $x \in \mathbb{R}^d$, the output of the model is $\widehat{y}^x$ defined as follows.

$$\begin{cases} h_0^{x,(L)} &= x, \\ h_k^{x,(L)} &= h_{k-1}^{x,(L)} + A_k^{(L)} h_{k-1}^{x,(L)}, \quad k = 1, \dots, L, \\ \widehat{y}_L^x &= h_L^x. \end{cases} \tag{2.1}$$

Our objective is to minimize the following unregularized mean-squared loss.

$$f^{(L)} \colon A^{(L)} \in \mathbb{R}^{L \times d \times d} \mapsto \mathbb{E}\left[\left\|Y - \widehat{y}_L^X\right\|_2^2\right]$$

We can rewrite the output as a linear function of the input:

$$\begin{aligned} \widehat{y}_L^x = h_L^{x,(L)} &= \left(I_d + A_L^{(L)}\right) h_{L-1}^{x,(L)} \\ &= \left(I_d + A_L^{(L)}\right) \cdots \left(I_d + A_1^{(L)}\right) h_0^{x,(L)} \\ &= \prod_{k=1}^{L} \left(I_d + A_k^{(L)}\right) x. \end{aligned}$$

Thus, we can rewrite the objective function as

$$\begin{aligned} f^{(L)}\left(A^{(L)}\right) &= \mathbb{E}\left[\left\|\left(R - \prod_{k=1}^{L}\left(I_d + A_k^{(L)}\right)\right) X + Z\right\|_2^2\right] \\ &= \left\|\left(R - \prod_{k=1}^{L}\left(I_d + A_k^{(L)}\right)\right)\Sigma^{1/2}\right\|_F^2 + \mathbb{E}\left[\|Z\|_2^2\right], \end{aligned} \tag{2.2}$$

where we use the independence of X and Z and standard matrix calculus in the second equality.

2.3 Description of the solutions

The global minimum of (2.2) is $f_{\min} = \mathbb{E}\left[\|Z\|_2^2\right] = \xi^2 d$ and is attained for multiple possible $A^{(L)}$. To compare the different minima, we introduce a tensor norm.

Definition 2.1. *Let $A^{(L)} \in \mathbb{R}^{L \times d \times d}$. The spectral norm of $A^{(L)}$ is defined as*

$$\left\|\left|A^{(L)}\right|\right\| \coloneqq \max_{k=1,\dots,L} \left\|A_k^{(L)}\right\|_2 .$$

The minimizers of (2.2) belong to one of the following three categories.

- *Atomic* solutions $A^{(L)} = (0_d, \dots, 0_d, R - I_d)$.

- The *symmetric* solution, defined as follows when R is positive semi-definite. We first orthogonally diagonalize R so that there exists an orthonormal matrix $U \in \mathbb{R}^{d\times d}$ and a diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_d)$, where $\lambda_i > 0$, such that $R = U\Lambda U^\top$. Then, we define
$$A_k^{*,(L)} = U\mathrm{diag}\left(\lambda_1^{1/L}, \dots, \lambda_L^{1/L}\right)U^\top - I_d.$$
We clearly have $f^{(L)}\left(A^{*,(L)}\right) = f_{\min}$, and as U is orthonormal, the spectral norm of $A^{*,(L)}$ is given by
$$|||A^{*,(L)}||| = \left\|A_1^{*,(L)}\right\|_2 = \left\|I_d - U\mathrm{diag}\left(\lambda_i^{1/L}\right)U^\top\right\|_2 = \max_{i=1,\dots,L}\left|\lambda_i^{1/L} - 1\right|.$$
We see that if L is big enough, e.g.
$$L \geq \gamma(R) \coloneqq \max\left\{\left|\log(\lambda_{\max}(R))\right|, \left|\log(\lambda_{\min}(R))\right|\right\},$$
then
$$|||A^{*,(L)}||| = \max_{i=1,\dots,L}\left|e^{\log(\lambda_i)/L} - 1\right| \leq \frac{3\gamma(R)}{L}.$$
The symmetric solution is also called the *minimum norm* solution as it minimizes the spectral norm of $A^{(L)}$ among all global minimizers of (2.2). Indeed, if $A^{(L)}$ is any global minima of (2.2), then
$$\left(1 + |||A^{*,(L)}|||\right)^L = \|R\|_2 = \left\|\prod_{k=1}^{L}\left(I_d + A_k^{(L)}\right)\right\|_2 \leq \left(1 + |||A^{(L)}|||\right)^L.$$
A similar construction applies when R is an arbitrary $d \times d$ matrix [68].

- *Hybrid* solutions spreading the total weight among a few $A_k^{(L)}$'s and setting the rest of them to zero.

We conclude that on $\mathbb{R}^{L\times d\times d}$, the landscape of $f^{(L)}$ has a lot of global minima, so points of vanishing gradient. However, if we restrict ourselves to a small ball $B_\tau^{(L)} = \left\{A^{(L)} : |||A^{(L)}||| \leq \tau\right\}$ around the origin, the squared norm of the gradient of $f^{(L)}$ is lower bounded by $f^{(L)} - f_{\min}$, called the *sub-optimality gap* of $f^{(L)}$. We say that such an $f^{(L)}$ satisfies a (local) *Polyak-Łojasiewicz* inequality [128]. It is not hard to see then that every critical point of $f^{(L)}$ in $B_\tau^{(L)}$ is a global optimum.[1]
More precisely, the lower bound on the norm of the gradient reads as follows.

[1] It was shown in [128] that the gradient descent dynamics of a loss function f satisfying a *global* Polyak-Łojasiewicz inequality converge linearly to a global optimum. This fact will be revisited more precisely in Section 2.4

Lemma 2.2 (Theorem 2.2 in [68]). *For any $A^{(L)} \in \mathbb{R}^{L\times d\times d}$ with $\left|\left|\left|A^{(L)}\right|\right|\right| \leq \tau$, we have*

$$\left\|\nabla f^{(L)}\left(A^{(L)}\right)\right\|_F^2 \geq 4L(1-\tau)^{2L-2}\lambda_{\min}(\Sigma)\left(f^{(L)}\left(A^{(L)}\right)-f_{\min}\right), \tag{2.3}$$

where $\|G\|_F := \left(\sum_{k=1}^L \|G_k\|_F^2\right)^{1/2}$ for $G \in \mathbb{R}^{L\times d\times d}$ and $\lambda_{\min}(\Sigma) > 0$ is the minimum singular value of Σ.

The proof follows after a straightforward gradient computation and can be found in [68].

Remark 2.3. *Note that the gradient lower bound (2.3) can be extended to a non-linear setting. It is shown in [16, Theorem 4] that if the input-output map $h\colon \mathbb{R}^d \to \mathbb{R}^d$ can be represented as a composition $h = h_L \circ \cdots \circ h_1$ of near-identity functions, i.e. where $\|h_i - \mathrm{id}_{\mathbb{R}^d}\|_\infty$ is small, then the loss function satisfies a local Polyak-Łojasiewicz inequality near the origin.*[2] *However, if we parametrize the function h_i via $h_i(x) = x + \delta^{(L)}\tanh\left(A_i^{(L)}x\right)$, then [16, Theorem 6] shows that $\left(\delta^{(L)}, A^{(L)}\right) = (0,0)$ is a critical point that is not always a global minimum of the loss function in the parameter space.*

2.4 Global convergence and scaling

In Section 2.3, we observed that the objective function (2.2) admits multiple different global minima. However, not all of them are equally likely to be reached by an optimization algorithm. Indeed, the solution such an algorithm finds is heavily dependent on its initialization scheme, its internal dynamics and the explicit regularization of the loss function, if any. Thus, given all the hyperparameters, it is important to have a principled way to study the convergence properties of the algorithm.

Remember that we are studying linear residual networks as a proxy for non-linear ones. Thus, in this section, we will focus on studying some variants of *gradient descent algorithms*, as they are the method of choice to optimize neural networks in practice.

Fix the depth $L \in \mathbb{N}^*$. We study the *continuous-time gradient descent* algorithm on the objective function (2.2). Namely, given an initialization scheme $A^{(L)}(0) = \left(A_k^{(L)}(0)\right)_{k=1,\cdots,L} \in \mathbb{R}^{L\times d\times d}$, we let the parameter $A^{(L)} \equiv A^{(L)}(t)$ evolve according to a

[2] The gradients are computed in the function space, where we use the notion of *Frechet derivative*.

velocity vector field equal to minus the gradient of the loss function $f^{(L)}$, where $t \geq 0$ represent the training time.

$$\frac{\mathrm{d}A_k^{(L)}}{\mathrm{d}t}(t) = -\nabla_{A_k} f^{(L)}\left(A^{(L)}(t)\right) \quad \text{for } k = 1, \ldots, L \quad \text{and} \quad t \geq 0. \tag{2.4}$$

We are interested in the following questions.

- Do we have asymptotic global minimization $f^{(L)}\left(A^{(L)}(t)\right) \to f_{\min}$ as $t \to \infty$?
- What is the convergence speed of $f^{(L)}\left(A^{(L)}(t)\right)$ to $f_{\min}$?
- Does the limit $A^{*,(L)} := \lim_{t\to\infty}\left(A_k^{(L)}(t)\right)_{k=1,\cdots,L}$ exist?
- What is the convergence speed of the parameters $A^{(L)}(t)$ to $A^{*,(L)}$ in the spectral norm?
- What is the scaling of the trained weights $A^{*,(L)}$ with respect to the depth L ?

To answer each of these questions, we follow the set-up of [152], where the so-called *ZAS initialization* is considered, i.e.

$$A_k^{(L)}(0) = 0_d \text{ for } k = 1, \ldots, L-1 \quad \text{and} \quad A_L^{(L)}(0) = -I_d. \tag{2.5}$$

The reason to initialize the last layer to $I_d + A_L^{(L)}(0) = 0_d$ is to ease the learning of negatively oriented matrices. Indeed, we have $\det\left(\prod_{k=1}^L\left(I_d + A_k^{(L)}\right)\right) = \prod_{k=1}^L \det\left(I_d + A_k^{(L)}\right) > 0$ if $\left|\left|\left|A^{(L)}\right|\right|\right|$ is small. To flip the sign of the determinant, one of the weight matrices has to move a distance $\Omega(1)$, which can take $\Omega(\exp(L))$ gradient descent iterations [152]. We will see in our Proposition 2.5 that with a ZAS initialization, the maximum distance traveled by a matrix $A_k^{(L)}$ is $\Theta(\log(L)^{1/2} \cdot L^{-1/2})$. To prove it, we first recall the results from [152] and prove the linear convergence of the loss function to $f_{\min}$.

Lemma 2.4 (Theorem 4.1 in [152]). *Fix $L \in \mathbb{N}^*$. Under the continuous-time gradient descent dynamics* (2.4) *with the ZAS initialization* (2.5), *we have*

$$f^{(L)}\left(A^{(L)}(t)\right) \leq f_{\min} + e^{-2\lambda_{\min}(\Sigma)t}\left(f^{(L)}\left(A^{(L)}(0)\right) - f_{\min}\right),$$

for all $t \geq 0$ and $R \in \mathbb{R}^{d\times d}$.

Proof. We present here the main ideas, the full proof is available in [152]. We omit the explicit dependence in L. For the sake of simplicity, we assume that the data is *whitened*, i.e. $\Sigma = I_d$. Define now $W_k := I_d + A_k$, and for $k_1 \leq k_2$,

$$W_{k_2:k_1} := \prod_{k=k_1}^{k_2} W_k = W_{k_2} W_{k_2-1} \cdots W_{k_1+1} W_{k_1}. \tag{2.6}$$

Then, $W_k(0) = I_d$ for $k = 1, \dots, L-1$ and $W_L(0) = 0_d$. The dynamics of the weights are kept unchanged as we are only adding a constant to the weights, so (2.4) reads

$$\frac{\mathrm{d}W_k}{\mathrm{d}t} = -\nabla_{W_k} \widetilde{f}(W), \quad \text{where} \quad \widetilde{f}(W) := f\left((W_k - I_d)_{k=1}^L\right). \tag{2.7}$$

The gradient can be written as follows.

$$\nabla_{W_k} \widetilde{f}(W) = \left(W_{L:k+1}\right)^\top \left(W_{L:1} - R\right) \left(W_{k-1:1}\right)^\top \tag{2.8}$$

Claim: For every $k = 1, \dots, L-1$, $D_k(t) := W_{k+1}^\top(t) W_{k+1}(t) - W_k(t) W_k^\top(t)$ is constant through time.
The claim can easily be shown by differentiating $D_k(t)$ with respect to t and by using (2.7). We thus deduce the relations

$$W_k W_k^\top = W_{k+1}^\top W_{k+1} \quad \text{for } k = 0, \dots, L-2, \tag{2.9}$$

$$W_{L-1} W_{L-1}^\top = I_d + W_L^\top W_L. \tag{2.10}$$

By direct computations, we also have

$$W_{L-1:1} W_{L-1:1}^\top = \left(I_d + W_L^\top W_L\right)^{L-1}. \tag{2.11}$$

Therefore,

$$\begin{aligned}\left\|\nabla_{W_L} \widetilde{f}(W)\right\|_F^2 &= \left\|(W_{L:1} - R) W_{L-1:1}^\top\right\|_F^2 \\ &\geq \lambda_{\min}\left(W_{L-1:1} W_{L-1:1}^\top\right) \|W_{L:1} - R\|_F^2 \geq 2\left(\widetilde{f}(W) - f_{\min}\right).\end{aligned}$$

Using this, we deduce that

$$\frac{\mathrm{d}}{\mathrm{d}t} \widetilde{f}(W(t)) \leq -\left\|\nabla_{W_L} \widetilde{f}(W(t))\right\|_F^2 \leq -2\left(\widetilde{f}(W(t)) - f_{\min}\right),$$

and the statement follows by Grönwall's lemma. □

We now address the following questions: do the individual weights $A_k^{(L)}(t)$ converge as $t \to \infty$ and if so, what is the scaling of $\lim_{t\to\infty} A_k^{(L)}(t)$ with respect to L.

Proposition 2.5. *Under the continuous-time gradient descent dynamics* (2.4) *with the ZAS initialization* (2.5), *the following holds.*

(i) $A^{(L)}(t)$ *converges linearly to some optimal weights* $A^{*,(L)} \in \mathbb{R}^{L\times d\times d}$ *as* $t \to \infty$ *in the tensor norm* $|||\cdot|||$.

(ii) $\Omega(L^{-1/2}) = \left\| I_d + A_L^{*,(L)} \right\|_2 = \mathcal{O}\left(L^{-1/2} (\log L)^{1/2}\right)$.

(iii) $\Omega\left(L^{-1}\right) = \left\| A_k^{*,(L)} \right\|_2 = \mathcal{O}\left(L^{-1/2} (\log L)^{1/2}\right)$ *for* $k = 1, \dots, L-1$.

Note that the following quantities

$$\left\| I_d + A_L^{*,(L)} \right\|_2 = \left\| A_L^{*,(L)} - A_L^{(L)}(0) \right\|_2 \quad \text{and} \quad \left\| A_k^{*,(L)} \right\|_2 = \left\| A_k^{*,(L)} - A_L^{(L)}(0) \right\|_2$$

are the total distances travelled by the parameters during the optimization.

Proof. We start by proving *(ii)*. Define as in the proof of Lemma 2.4 the matrices $W_k \coloneqq I_d + A_k$ for $k = 1, \dots, L$. The input-output map of the linear residual network then reads $x \mapsto W_{L:1}\, x$, where the notation $W_{L:1}$ is defined in (2.6). We use (2.11) to compute

$$\begin{aligned} W_{L:1} W_{L:1}^\top &= W_L W_{L-1:1} W_{L-1:1}^\top W_L^\top \\ &= W_L \left(I_d + W_L^\top W_L \right)^{L-1} W_L^\top \\ &= \sum_{k=0}^{L-1} \binom{L-1}{k} \left(W_L W_L^\top \right)^{k+1}. \end{aligned}$$

Hence, if we let $(\nu_i)_{i=1,\dots,d}$ be the (positive) eigenvalues of $W_L W_L^\top$, we have $\operatorname{tr}\left((W_L W_L^\top)^k\right) = \sum_{i=1}^d \nu_i^k$ and

$$\begin{aligned} \| W_{L:1} \|_F^2 = \operatorname{tr}\left(W_{L:1} W_{L:1}^\top \right) &= \sum_{k=0}^{L-1} \binom{L-1}{k} \operatorname{tr}\left((W_L W_L^\top)^{k+1} \right) \\ &= \sum_{k=0}^{L-1} \binom{L-1}{k} \sum_{i=1}^d \nu_i^{k+1} = \sum_{i=1}^d \nu_i \left(1 + \nu_i\right)^{L-1}. \end{aligned} \tag{2.12}$$

Now, choose $t_0 > 0$ big enough so that $f(A(t)) - f_{\min} < \frac{\|R\|_F^2}{4}$ for all $t > t_0$. Using (2.12), for $t > t_0$, we get the following bound.

$$\frac{9}{4} \|R\|_F^2 \geq \left(\|R - W_{L:1}(t)\|_F + \|R\|_F \right)^2 \geq \|W_{L:1}(t)\|_F^2 \geq \nu_{\max}(t) \left(1 + \nu_{\max}(t)\right)^{L-1}, \tag{2.13}$$

where $\nu_{\max} := \max_{i=1,\dots,d} \nu_i = \|W_L\|_2^2$ is the squared spectral norm of W_L. Similarly, we have

$$\frac{1}{4}\|R\|_F^2 \leq \left(\|R\|_F - \|R - W_{L:1}(t)\|_F\right)^2 \leq \|W_{L:1}(t)\|_F^2 \leq d\,\nu_{\max}(t)\,(1+\nu_{\max}(t))^{L-1}.$$

Thus, we deduce the following lower bound:

$$\nu_{\max}(t) \geq \log(1+\nu_{\max}(t)) \geq \log\left((1+\nu_{\max}(t))^L\right)L^{-1} \geq \log\left(\frac{\|R\|_F^2}{4d}\right)L^{-1} =: c_0 L^{-1}$$

Similarly, we have that for $L \geq c_0$, using (2.13),

$$(1+\nu_{\max}(t))^L \leq 2c_0^{-1}L\nu_{\max}(t)(1+\nu_{\max}(t))^L \leq \frac{9}{2c_0}\|R\|_F^2\,L \tag{2.14}$$

This means, for each $t \geq 0$,

$$\nu_{\max}(t) = \Omega(L^{-1}) \quad \text{and} \quad \nu_{\max}(t) = \mathcal{O}(L^{-1}\log L). \tag{2.15}$$

We conclude *(ii)* by noting that $\|W_L^*\|_2^2 = \lim_{t\to\infty}\nu_{\max}(t)$.

Next, we prove (iii). Observe first that $\|W_k(u)\|_2^2 = 1 + \|W_L(u)\|_2^2$ for each $k < L$ by (2.9) and (2.10). Therefore, there exist some constants $c_1, c_2 > 0$ such that

$$1 + c_1 L^{-1} \leq \|I_d + A_k(t)\|_2^2 \leq 1 + c_2 L^{-1}\log L$$

for $t > t_0$ and $L > c_0$. Hence

$$\|A_k(t)\|_2 \geq \|I_d + A_k(t)\|_2 - 1 = \left(1 + c_1 L^{-1}\right)^{1/2} - 1 \geq \frac{c_1}{3}L^{-1},$$

where the last inequality holds for $L > c_1/3$. Hence, in the limit $t \to \infty$, we get $\|A_k^*\|_2 = \Omega(L^{-1})$ by (2.15). Similar calculations give the corresponding upper bound.

Finally, we prove (i). For $t_0 < s < t$ and $k = 1, \dots, L-1$, we get by (2.4)

$$\begin{aligned}\|A_k(t) - A_k(s)\|_F^2 &= \left\|\int_s^t \frac{\mathrm{d}}{\mathrm{d}u}A_k(u)\mathrm{d}u\right\|_F^2 \\ &\leq \int_s^t \left\|\nabla_{A_k} f^{(L)}\left(A^{(L)}(t)\right)\right\|_F^2 \mathrm{d}u \\ &\leq 4\int_s^t \|W_L(u)\|_2^2\,(f(A(u)) - f_{\min})\prod_{j\neq k,L}\|W_j(u)\|_2^2\,\mathrm{d}u,\end{aligned}$$

where the second inequality holds by the gradient derivation in [68] and Lemma D.1. Next, we use the fact that $\|W_j(u)\|_2^2 = 1 + \|W_L(u)\|_2^2$ for $j < L$ by (2.9) and (2.10).

From the upper bound above, we thus get

$$\begin{aligned}
\|A_k(t) - A_k(s)\|_F^2 &\leq 4 \int_s^t \left(1 + \nu_{\max}(u)\right)^{L-2} \nu_{\max}(u) \left(f(A(u)) - f_{\min}\right) \mathrm{d}u \\
&\leq 9 \|R\|_F^2 \int_s^t \left(f(A(u)) - f_{\min}\right) \mathrm{d}u \\
&\leq \frac{9}{2} \|R\|_F^2 \left(f(A(0)) - f_{\min}\right) \left(e^{-2s} - e^{-2t}\right).
\end{aligned}$$

The second inequality is due to the upper bound (2.13) and the third to Lemma 2.4. A similar bound holds for A_L:

$$\begin{aligned}
\|A_L(t) - A_L(s)\|_F^2 &\leq 4 \int_s^t \left(1 + \nu_{\max}(u)\right)^{L-1} \left(f(A(u)) - f_{\min}\right) \mathrm{d}u \\
&\leq 9 \|R\|_F^2 \int_s^t \nu_{\max}(u)^{-1} \left(f(A(u)) - f_{\min}\right) \mathrm{d}u \\
&\leq \frac{9}{2c_0} \|R\|_F^2 \left(f(A(0)) - f_{\min}\right) \left(e^{-2s} - e^{-2t}\right) L.
\end{aligned}$$

Thus, for a fixed L, we have linear convergence of $A(t)$ to the global minima A^* when $t \to \infty$ in the Frobenius norm, so in the tensor norm as well. □

Remark 2.6. *If to optimize the loss function $f^{(L)}$ we use discrete-time gradient descent instead, i.e. $A_k^{(L)}(t+1) = A_k^{(L)}(t) - \eta^{(L)}(t) \nabla_{A_k} f^{(L)}\left(A^{(L)}(t)\right)$, observe that the parameters $A_k^{(L)}$, $k \neq L$, are $\Omega\left(L^{-1}\right)$ by Proposition 2.5 whereas the gradients $\nabla_{A_k} f^{(L)}\left(A^{(L)}\right)$ are $\Theta(1)$. Thus, the sum of learning rates $\sum_{t=0}^{\infty} \eta^{(L)}(t)$ has to scale like $\Omega\left(L^{-1}\right)$ for an optimal learning process. This observation is supported by the experiments in Section 2.6.*

2.5 Existence of a scaling limit as the depth increases and connection to linear neural ODE

In Proposition 2.5, we study the scaling of the optimal weights with respect to the depth under the ZAS initialization (2.5). We observe that the optimal weights $A_k^{*(L)}$, $k < L$, scale differently than $A_L^{*(L)}$, making it unlikely that a scaling limit exist. In general, the following design choices influence the scaling of the weights and whether a scaling limit exist.

- The explicit scaling
- The initialization scheme

- The learning rate scaling

For example, Theorem 1 in [138] proves that with an explicit $1/L$ in front of the matrices $A_k^{(L)}$, a learning rate $\eta_L(t)$ scaling linearly with L, and initial weights $A_k^{(L)}(0)$ such that $L\left(A_{\lfloor Ls\rfloor}^{(L)}(0) - \psi_s(0)\right) \to 0$ for each $s \in [0,1]$ as $L \to \infty$, then the weights trained by gradient descent also admit a scaling limit: $A_{\lfloor Ls\rfloor}^{(L)}(t) \to \psi_s(t)$ as $L \to \infty$. Furthermore, the hidden states $h_{\lfloor Ls\rfloor}^{x,(L)}(t)$ converge uniformly to $H_s^x(t)$ as $L \to \infty$, which solves the following linear neural ODE:

$$\frac{\mathrm{d}}{\mathrm{d}s} X_s^x(t) = \psi_s(t) X_s^x(t) \quad \text{for } s \in [0,1]\,, \quad X_0^x(t) = x.$$

The key difference with our framework is that the weights $A_k^{(L)}$ and the rescaled gradients $\eta^{(L)} \nabla_{A_k} f^{(L)}\left(A^{(L)}\right)$ are of the same order $\mathcal{O}(1)$ with respect to L, in contrast to our framework.

2.6 Numerical examples

We illustrate the above results with some numerical experiments. We fix $d = 10$ and a matrix $R \in \mathbb{R}^{d\times d}$ whose eigenvalues are taken uniformly in $[0.1, 10]$ so that the condition number of R is large. We generate $N = 1024$ i.i.d samples $\{x_i\}_{i=1}^N$ from the distribution $\mathcal{D} = \mathcal{N}(0, I_d)$, and we compute the targets $y_i = Rx_i + z_i$, where $z_i \sim \mathcal{N}(0, \xi I_d)$ and $\xi = 0.1$. We train a linear residual network (2.1) using stochastic gradient descent with a learning rate $\eta^{(L)}$ and batch size $B = 32$, meaning that we choose the samples $\mathcal{B}_k \subset \{1, \dots, N\}$ uniformly at random at iteration k, with $|\mathcal{B}_k| = B$. The parameter updates thus read, for $t \geq 1$,

$$A^{(L)}(t) = A^{(L)}(t-1) - \eta^{(L)} \cdot \nabla_A \left(\frac{1}{B} \sum_{i \in \mathcal{B}_k} \left\| y_i - \prod_{k=1}^{L} \left(I_d + A_k^{(L)}\right) \cdot x_i \right\|_2^2 \right).$$

We choose the learning rate in an adaptive way. First, we train the network at the lowest depth $L_0 = 2$ with an initial learning rate $\eta^{(L_0)} = 0.05$. We perform SGD iterations until either the training loss $f_N^{(L_0)}\left(A^{(L_0)}(t)\right)$ drops below $f_{\min} + \varepsilon$, where $\varepsilon = 10^{-2}$, or the maximum number of iterations $t_{\max} = 3200$ is reached. In the former case, we increase the depth and the network is reinitialized. In the latter case, we decrease the learning rate by 30%, meaning that the new learning rate will be $\eta^{(L_0)} \leftarrow 0.7 \cdot \eta^{(L_0)}$, and we train the network with the same depth, but with the new learning rate. We continue until the network with the largest depth $L_{\max} = 945$ is trained.

We first present the results when we initialize the network weights using ZAS initialization (2.5). In Figure 2.1, we observe that the last layer is staying at a constant order while increasing the depth, whereas the other layers' norms decrease a bit faster than $1/L$. In Figure 2.2, we observe that the learning rate has to decrease linearly with the depth to achieve convergence of the training loss, in accordance to Remark 2.6

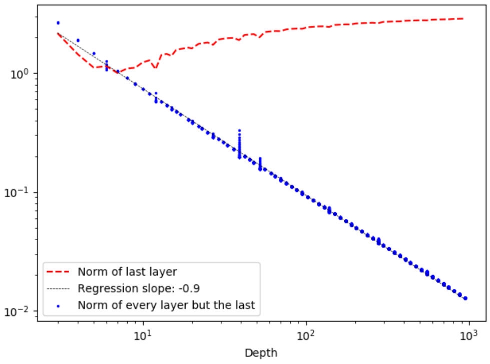

Figure 2.1: Frobenius norm of the trained weights at different depths with the ZAS initialization (2.5). In red: norm of the last layer. In blue: norms of all the other layers.

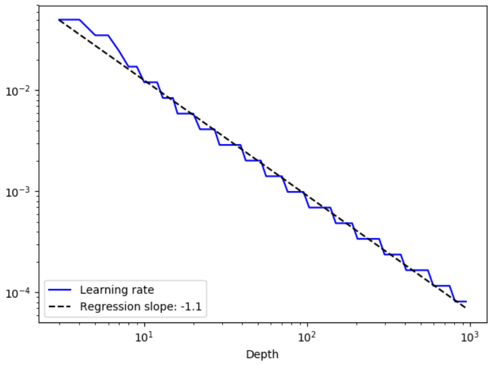

Figure 2.2: Learning rate $\eta^{(L)}$ used in SGD to train the weights at different depths with the ZAS initialization (2.5). The learning rate is updated according to the schedule described in Section 2.6.

However, with a different initialization scheme, we observe a different behaviour. Indeed, with the Xavier[3] initialization [59], the norms of the trained weights $A_k^{*,(L)}$ are all of the order of $1/L$, see Figure 2.3. Observe as well in Figure 2.4 that the learning rate has to decrease linearly with the depth as well to achieve convergence, after a small initial plateau.

[3] $A_k^{(L)}(0)_{mn}$ are independent samples taken from a centered Gaussian distribution with standard deviation L^{-1}.

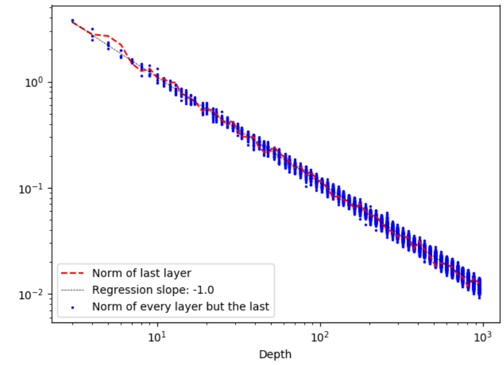

Figure 2.3: Frobenius norm of the trained weights at different depths with the Xavier initialization. In red: norm of the last layer. In blue: norms of all the other layers.

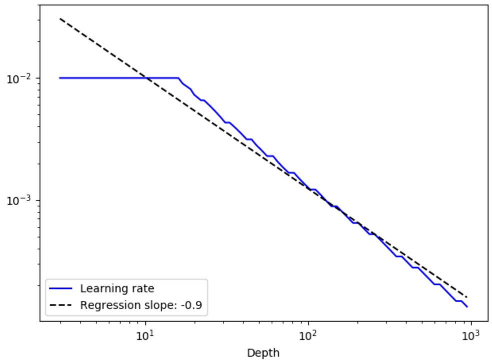

Figure 2.4: Learning rate $\eta^{(L)}$ used in SGD to train the weights at different depths with the Xavier initialization. The learning rate is updated according to the schedule described in Section 2.6.

Both examples show that gradient descent finds a solution close to its initialization. This observation is one of the main ideas driving convergence proofs in the non-linear case, as will be seen in Chapter 4.

Chapter 3

Scaling properties of deep residual networks

3.1 Introduction

Residual networks, or ResNets, are multilayer neural network architectures in which a *skip connection* is introduced at every layer ([73]). This allows very deep networks to be trained by circumventing vanishing and exploding gradients, mentioned in [20]. The increased depth in ResNets has lead to commensurate performance gains in applications ranging from speech recognition [74, 161] to computer vision [73, 78].

A residual network with L layers may be represented as

$$h_{k+1}^{(L)} = h_k^{(L)} + \delta_k^{(L)} \sigma_d \left(A_k^{(L)} h_k^{(L)} + b_k^{(L)} \right), \tag{3.1}$$

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, $\sigma \colon \mathbb{R} \to \mathbb{R}$ a nonlinear 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 $\delta_k^{(L)}$ trainable network weights for $k = 0, \ldots, L-1$.

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 [67] and [28], is to view (3.1) as a discretization of a system of ordinary differential equations

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \sigma_d \left(\overline{A}_t H_t + \overline{b}_t \right), \tag{3.2}$$

where $\overline{A} \colon [0,1] \to \mathbb{R}^{d \times d}$ and $\overline{b} \colon [0,1] \to \mathbb{R}^d$ are appropriate smooth functions and $H(0) = x$. This may be justified ([148]) by assuming that

$$\delta^{(L)} \sim 1/L, \quad A_k^{(L)} \sim \overline{A}_{k/L}, \quad b_k^{(L)} \sim \overline{b}_{k/L} \tag{3.3}$$

as L increases. Such models, named neural ordinary differential equations or neural ODEs [28, 46], have motivated the use of optimal control methods to train ResNets [47].

However, the precise link between deep ResNets and the neural ODE (3.2) is unclear: in practice, the weights $A^{(L)}$ and $b^{(L)}$ result from training, yet the validity of the scaling assumptions (3.3) for trained weights is far from obvious. 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 [13]. Also, there is no guarantee that weights obtained through training have a non-zero limit which depends smoothly on the layer, as (3.3) would require. In fact, for ResNet architectures used in practice, empirical evidence points to the contrary [37]. 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.

Contributions. We systematically investigate the scaling behavior of trained networks weights and examine in detail the consequence of this scaling for the asymptotic properties of ResNets as the number of layers increases. We first show, through detailed numerical experiments, the existence of scaling regimes for trained weights markedly different from those implicitly assumed in the neural ODE literature. We study the convergence of the hidden state dynamics in these scaling regimes, showing that one may obtain an ODE, a stochastic differential equation (SDE) or neither of these. More precisely, we show strong convergence of the hidden state dynamics to a limiting ODE or SDE, by viewing the discrete hidden state dynamics as a "nonlinear Euler scheme" of the limiting equation. At a mathematical level, we extend the convergence analysis of Higham et al. [75] for discretization schemes of time-homogeneous (Markov) diffusions to a class of nonlinear approximations for Itô processes with bounded coefficients.

In particular, our findings point to the existence of a "diffusive regime" in which the deep network limit is described by a class of stochastic differential equations (SDEs). These novel findings on the relation between ResNets and neural ODEs complement previous work [148, 48, 56, 123, 138]. Finally, we derive the corresponding scaling limit for the backpropagation dynamics. The results we obtain are different from previous ones on asymptotics of ResNets [28, 67, 106], and correspond to a different scaling regime which is relevant for trained weights in practical settings.

In particular, in the diffusive regime we find a limit different from the "Neural SDE" literature [98]. Indeed, we observe that the Jacobian of the output with respect to the hidden states depends on hidden states across all levels, so may *not* be directly expressed as the solution of a forward or backward stochastic differential equation, as proposed in [98]. However, in Section 3.5 we obtain a representation for the asymptotics of the backpropagation dynamics in terms of an auxiliary forward SDE.

Outline. Section 3.2 describes the various scaling regimes for trained weights evidenced in [37] and the methodology for studying this scaling behaviour in the deep network limit. In Section 3.3, we report detailed numerical experiments on 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 (3.3). In Section 3.4, we show that under these scaling regimes, the dynamics of the the hidden state may be described in terms of a class of ordinary or stochastic differential equations, different from the neural ODEs studied in [28, 67, 106]. In Section 3.5, we derive the large depth limit of the backpropagation dynamics under each scaling regime.

Notations. Let $\|v\|$ denote the Euclidean norm of a vector v. For a matrix M, denote $M^\top$ its transpose, $\operatorname{diag}(M)$ its diagonal vector, $\operatorname{tr}(M)$ its trace and $\|M\|_F = \sqrt{\operatorname{tr}(M^\top M)}$ its Frobenius norm. Denote $\lfloor x \rfloor$ the integer part of a real number x. Let $\mathcal{N}(m, \Sigma)$ denote the Gaussian distribution with mean m and (co)variance Σ, $\otimes$ denote the tensor product, and $\mathbb{R}^{d,\otimes n} = \mathbb{R}^d \times \cdots \times \mathbb{R}^d$ (n times). Define the vectorisation operator by $\operatorname{vec}\colon \mathbb{R}^{d_1 \times \cdots \times d_n} \to \mathbb{R}^{d_1 \cdots d_n}$, and let $\mathbb{1}_S$ be the indicator function of a set S. $\mathcal{C}^0$ is the space of continuous functions, for $\nu \geq 0$, $\mathcal{C}^\nu$ is the space of ν-Hölder continuous functions, and $\mathcal{H}^1$ is the Sobolev space of order 1.

3.2 Scaling regimes

We start by providing a framework for describing the scaling regimes for trained network weights, as identified in numerical experiments on deep ResNets [37].

3.2.1 Scaling regimes for trained network weights

As described in Section 3.1, the neural ODE limit assumes

$$\delta^{(L)} \sim \frac{1}{L} \quad \text{and} \quad A^{(L)}_{\lfloor Lt \rfloor} \xrightarrow{L\to\infty} \overline{A}_t, \quad b^{(L)}_{\lfloor Lt \rfloor} \xrightarrow{L\to\infty} \overline{b}_t \tag{3.4}$$

for $t \in [0,1]$, where $\overline{A}\colon [0,1] \to \mathbb{R}^{d\times d}$ and $\overline{b}\colon [0,1] \to \mathbb{R}^d$ are smooth functions [148]. Our numerical experiments, detailed in Section 3.3, show that the norm of the weights generally shrinks as L increases (see for example Figures 3.2 and 3.4), so one cannot expect the above assumption to hold, unless weights are renormalized in some way. We consider here a more general assumption which includes (3.4) but allows for shrinking weights.

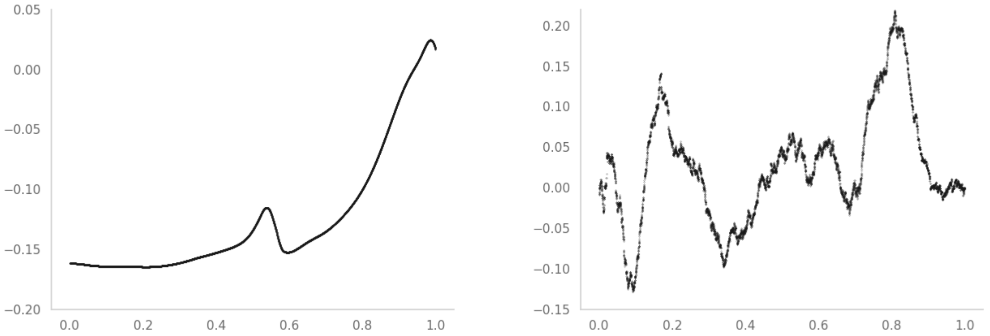

Figure 3.1: Trained weights as a function of k/L for $k = 0, \dots, L$ and $L = 9100$. Left: rescaled weights $L^{\beta} A^{(L)}_{k,(0,0)}$ for a tanh network with $\beta = 0.2$. Right: cumulative sum $\sum_{j=0}^{k-1} A^{(L)}_{j,(0,0)}$for a ReLU network. Note that each $A^{(L)}_{k,(0,0)} \in \mathbb{R}$.

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*

$$\forall s \in [0,1], \qquad \overline{A}_s = \lim_{L\to\infty} L^{\beta}\, A^{(L)}_{\lfloor Ls \rfloor}. \tag{3.5}$$

These renormalized weights do converge to a continuous function of the layer in some cases, as shown in Figure 3.1 (top) which displays a ResNet (3.1) with fully connected layers and tanh activation function, without explicit regularization (see Section 3.3.2).

Yet, it is not the case that network weights always converge to a smooth function of the layer, even after rescaling. Indeed, 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. In this case, the *cumulative sum* $\sum_{j=0}^{k-1} A^{(L)}_j$ 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 3.1 (bottom) 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 a different scaling regime where the weights $A^{(L)}_k$ are represented as the *increments* of a continuous function W^A, i.e. the *cumulative sum* of the weights may converges to a limit but not the weight themselves. We also allow for a *trend* term as in Scaling regime 1.

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* $W^A_0 = 0$ *and*

$$A^{(L)}_k = L^{-\beta}\overline{A}_{k/L} + W^A_{(k+1)/L} - W^A_{k/L}. \tag{3.6}$$

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

$$L^{\beta-1}\sum_{k=0}^{\lfloor Ls\rfloor -1} A_k^{(L)} = L^{-1}\sum_{k=0}^{\lfloor Ls\rfloor -1} \overline{A}_{k/L} + L^{\beta-1}W^A_{\lfloor Ls\rfloor/L} \overset{L\to\infty}{\to} \int_0^s \overline{A}_r \mathrm{d}r. \tag{3.7}$$

The integral of $\overline{A}$ is thus uniquely determined by the weights $A_k^{(L)}$, so $\overline{A}$ can be obtained by discretization and W^A by fitting the residual error in (3.7). In addition, Scaling Regimes 1 and 2 are mutually exclusive since Scaling regime 2 requires W^A to be non-zero.

Remark 3.1. *In the case of independent Gaussian weights*

$$A^{(L)}_{k,mn} \overset{iid}{\sim} \mathcal{N}\left(0, L^{-1}d^{-2}\right) \quad \textit{and} \quad b^{(L)}_{k,n} \overset{iid}{\sim} \mathcal{N}\left(0, L^{-1}d^{-1}\right),$$

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

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

which is a special case of Scaling regime 2.

This remark shows that Scaling regime 2 corresponds to a 'diffusive' regime.

3.2.2 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 3.1 several quantities which may be viewed as discrete versions of various (semi-)norms used to measure the smoothness of functions.

Table 3.1: Quantities associated to a tensor $A^{(L)} \in \mathbb{R}^{L\times d\times d}$.

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

3.3 Scaling behavior of trained weights: numerical experiments

We now report on detailed numerical experiments to 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.

3.3.1 Methodology

We underline that Scaling Regimes 1 and 2 are mutually exclusive since Scaling regime 2 requires W^A to be non-zero. In order to examine whether one of these scaling regimes, or neither, holds for the trained weights $A^{(L)}$ and $b^{(L)}$, we proceed as follows.
Step 1: First, to obtain the scaling exponent $\beta \in [0, 1)$, note that under Scaling regime 2,

$$L^{\beta-1}\sum_{k=1}^{L} A_k^{(L)} = \frac{1}{L}\sum_{k=1}^{L} \overline{A}_{k/L} + L^{\beta-1}W_1^A \qquad \overset{L\to\infty}{\to} \int_0^1 \overline{A}_s \mathrm{d}s.$$

Hence, we perform a logarithmic regression of the cumulative sum norm of $A^{(L)}$ with respect to L, and the rate of increase of $\sum_{k=1}^{L} A_k^{(L)}$ as $L \to \infty$ is $1 - \beta$.
Step 2: After identifying the correct scale $L^{-\beta}$ for the weights, we compute the β-scaled norm of increments of $A^{(L)}$ to check whether they satisfy Scaling regime 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 Scaling regime 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 Scaling regime 1 with a Hölder-continuous limit function $\overline{A} \in \mathcal{C}^{\nu}([0, 1], \mathbb{R}^{d\times d})$.
Step 3: To discriminate between Scaling regimes 1 and 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 (3.7). The presence of non-negligible noise term W^A favors Scaling regime 2.
Step 4: Finally, we estimate the regularity of the term W^A under Scaling regime 2. If W^A has *diffusive* behavior, as in the example of i.i.d. random weights, then its quadratic variation tensor defined by

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

has a finite limit as $L \to \infty$. Hence, using (3.6) and Cauchy-Schwarz, we obtain

$$\left|\!\left|\!\left| \left[W^A\right]_s \right|\!\right|\!\right| \leq 2 \cdot \lim_{L\to\infty} \sum_{k=0}^{\lfloor Ls \rfloor -1} \left\| A_k^{(L)} \right\|_F^2 + L^{1-2\beta} \left\| \overline{A} \right\|_{L^2}^2 \tag{3.8}$$

where $|\!|\!|\cdot|\!|\!|$ is the Hilbert-Schmidt norm. As $\overline{A}$ is continuous on a compact domain, its L^2 norm is finite. Hence, if we have $\beta \geq \frac{1}{2}$, 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.

We follow all of the above steps for $b^{(L)}$ as well. Note that the scaling exponent β may not be the same for $A^{(L)}$ and $b^{(L)}$.

Remark 3.2. *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.*

3.3.2 Results for fully-connected layers

We first consider the case where the network layers are fully-connected. We consider the network architecture (3.1) 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.

We consider two data sets. 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:

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

where $g_d(z, k, K) \coloneqq \sin(5k\pi/K)z + \cos(5k\pi/K)\mathbb{1}_d$. The targets y_i are defined as $y_i = z_K^{x_i} / \left\| z_K^{x_i} \right\|$. The motivation behind this low-dimensional dataset is to be able to train very deep residual networks and to be sure that there exists at least a (sparse) optimal solution.

The second dataset is a low-dimensional embedding of the MNIST handwritten digits dataset [94]. 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) using batches of size B on the mean-square loss and a constant learning rate η, until the loss falls below ϵ, or when the maximum number of updates $T_{\max}$ is reached. We repeat the experiments for depths L varying from $L_{\min}$ to $L_{\max}$. Details are given in Appendix A.

Results. For the case of a tanh activation (i), we observe in Figure 3.2 that for both datasets, $\delta^{(L)} \sim L^{-0.7}$ clearly decreases as L increases, and $A^{(L)}$ decreases slightly when L increases. We deduce that $\beta = 0.3$ for the MNIST dataset and $\beta = 0.2$ for the synthetic dataset.

We use these results to identify the scaling behavior of $A^{(L)}$. We observe in Figure 3.3 (left) that the β-scaled norm of increments of $A^{(L)}$ decreases like $L^{-1/2}$, suggesting that Scaling regime 1 holds, with $\overline{A}$ being $1/2-$Hölder continuous. This is confirmed in Figure 3.3 (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**.

Regarding the case of a ReLU activation function (ii), we observe in Figure 3.4 (left) that the trend part of the residual connection $\left|\delta^{(L)}\right| A^{(L)}$ scales like $L^{-0.8}$ for the synthetic dataset and like $L^{-0.9}$ for the MNIST dataset. We see in Figure 3.4

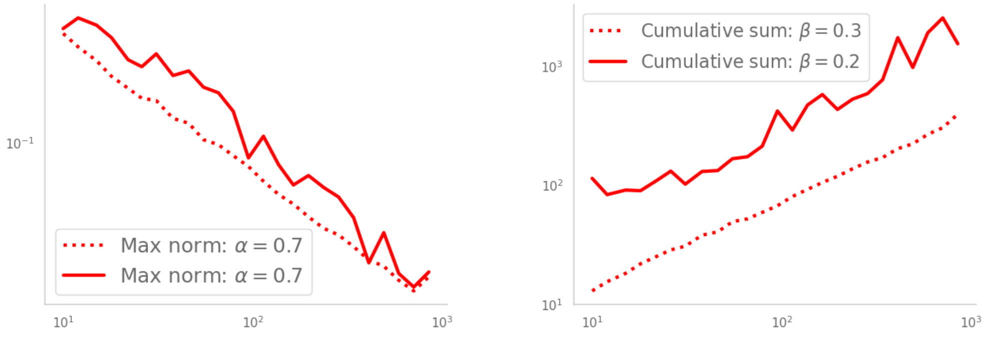

Figure 3.2: Scaling for tanh activation and $\delta^{(L)} \in \mathbb{R}$. Left: Maximum norm of $\delta^{(L)}$ with respect to L. Right: Cumulative sum norm of $A^{(L)}$ with respect to L. The dashed lines are for the synthetic data and the solid lines are for MNIST. The plots are in log-log scale.

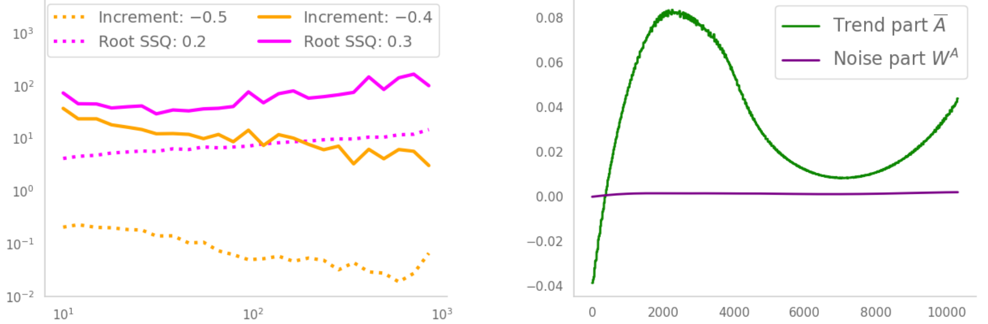

Figure 3.3: Identification of scaling behavior in the case of tanh activation and $\delta^{(L)} \in \mathbb{R}$. Left: log-log plot of root sum of squares of $A^{(L)}$ (pink) and the β-scaled norm of increments of $A^{(L)}$ (orange). Dashed lines are for the synthetic data and the solid lines are for 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 (3.6), for the synthetic dataset.

(right) that keeping the sign of $\delta_k^{(L)}$ is important, as the sign oscillates considerably throughout the network depth $k = 0, \ldots, L-1$.

Figure 3.5 (left) shows that the β-scaled norm of increments diverges as the depth increases. This suggests that there exists a noise part W^A. Following (3.8), the fact that the root sum of squares of $\left|\delta^{(L)}\right| A^{(L)}$ is upper bounded as $L \to \infty$ and $\beta \geq 1/2$ implies that W^A has finite quadratic variation. These claims are also supported by Figure 3.5 (right): there is a non-zero trend part $\overline{A}$, and a non-negligible noise part W^A.

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

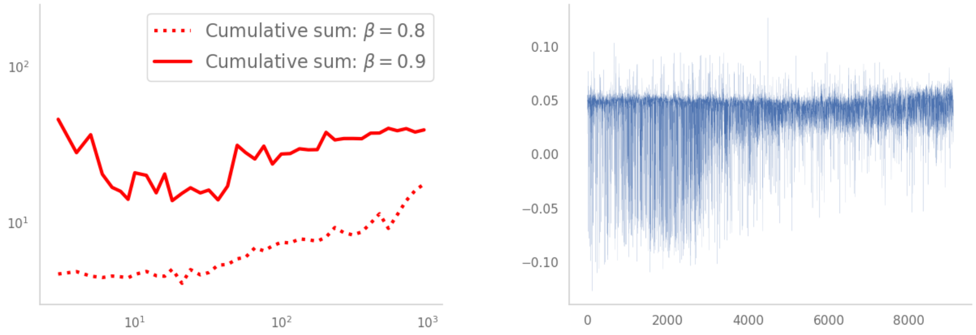

Figure 3.4: Scaling for ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$. Left: Cumulative sum norm of $|\delta^{(L)}|A^{(L)}$ with respect to L, in log-log scale. Right: trained values of $\delta_k^{(L)}$ as a function of k, for $L = 9100$ and for the synthetic dataset.

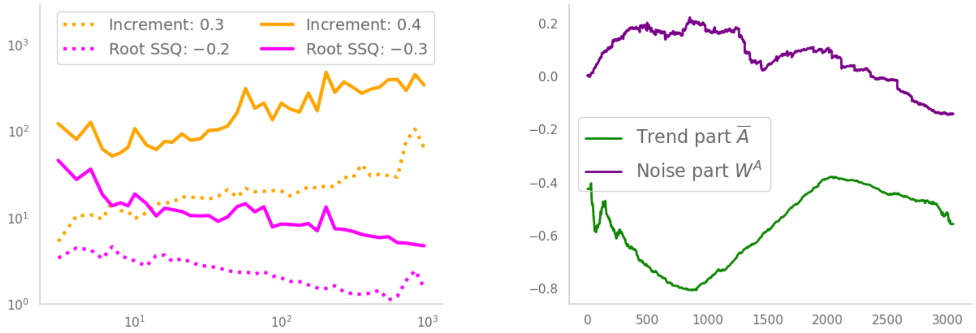

Figure 3.5: ReLU activation and scalar $\delta_k^{(L)}$. Left: in pink we plot in log-log scale the root sum of squares of $|\delta^{(L)}|A^{(L)}$, and in orange the β-scaled norm of increments of $|\delta^{(L)}|A^{(L)}$. The dashed lines are for the synthetic data and the solid lines for MNIST. Right: Decomposition of the trained weights $|\delta^{(L)}|\, A_{k,(7,7)}^{(L)}$ with the trend part $\overline{A}$ and the noise part W^A for $L = 10321$, as defined in (3.6), for the synthetic dataset.

Scaling behavior of $b^{(L)}$ are shown for the tanh case in Figure 3.6 and for the ReLU case in Figure 3.7. We observe that the cumulative sum norm, the scaled norm of the increments and the root sum of squares of $b^{(L)}$ scales in the same way as $A^{(L)}$ as the depth L increases. In particular, the scaling exponent β for $b^{(L)}$ is equal to the scaling exponent of $A^{(L)}$, justifying the setup considered in Section 3.2.

Importance of the stochastic term W^A. It is legitimate to ask whether the noise term W^A plays a significant role in the output accuracy of the network. To test this, we create a residual network with denoised weights $\widetilde{A}_k^{(L)} := L^{-\beta}\overline{A}_{k/L}$, compute its training error and we compare it to the original training error. We observe in Figure 3.8 (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

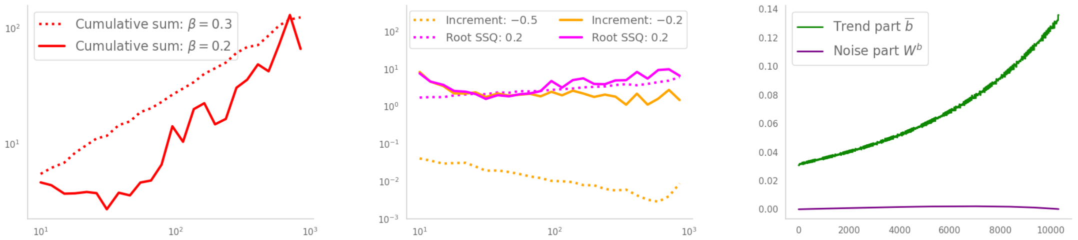

Figure 3.6: Scaling behavior for $b^{(L)}$ with tanh activation and scalar $\delta^{(L)}$. Left: cumulative sum norm of $b^{(L)}$ with respect to L, in log-log scale. Middle: the root sum of squares of $b^{(L)}$ in pink and the $\beta-$scaled norm of increments of $b^{(L)}$ in orange, in log-log scale. The dashed lines are for the synthetic data and the solid lines are for MNIST. Right: Decomposition of the trained weights $b_{k,5}^{(L)}$ with the trend part $\bar{b}$ and the noise part W^b for $L = 10321$, as defined in (3.6), for the synthetic dataset.

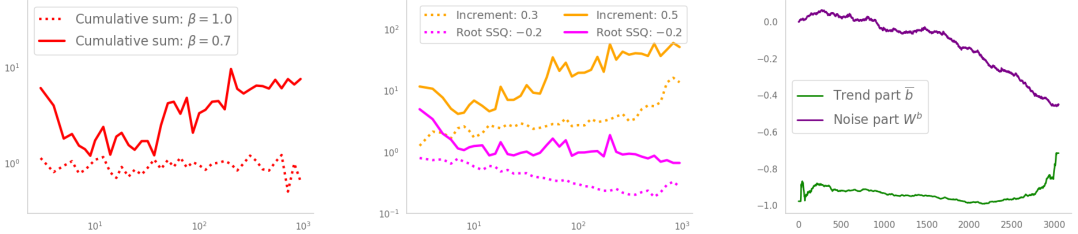

Figure 3.7: Scaling and hypothesis verification for $b^{(L)}$ with ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$. Left: cumulative sum norm of $|\delta^{(L)}|b^{(L)}$ with respect to L, in log-log scale. Middle: the root sum of squares of $|\delta^{(L)}|b^{(L)}$ in pink and the $\beta-$scaled norm of increments of $|\delta^{(L)}|b^{(L)}$ in orange, in log-log scale. The dashed lines are for the synthetic data and the solid lines for MNIST. Right: Decomposition of the trained weights $|\delta^{(L)}|\, b_{k,6}^{(L)}$ with the trend part $\bar{b}$ and the noise part W^b for $L = 10321$, as defined in (3.6), for the synthetic dataset.

above the original training loss, meaning that the noise part W^A plays a significant role in the accuracy of the trained network.

Sensitivity of α and β with respect to the hyperparameters. The values of α and β stem from the trained weights, which are themselves a function of the initialization and the training algorithm. We are using stochastic gradient descent, and the most significant hyperparameters of SGD are the learning rate η and the batch size B.

Hence, we report the value α and β found for the tanh and trainable δ architecture on the synthetic data with different batch sizes $B \in \{8, 32, 128\}$ and learning rates $\eta \in \{0.01, 0.003, 0.001\}$, with 5 different realizations for the initialization. We report the average values of α and β for 5 different seeds in Table 3.2 below.

Table 3.2: Average value of α (left) and β (centre) for the trained weights, over 5 random initialization. η is the learning rate, B the batch size.

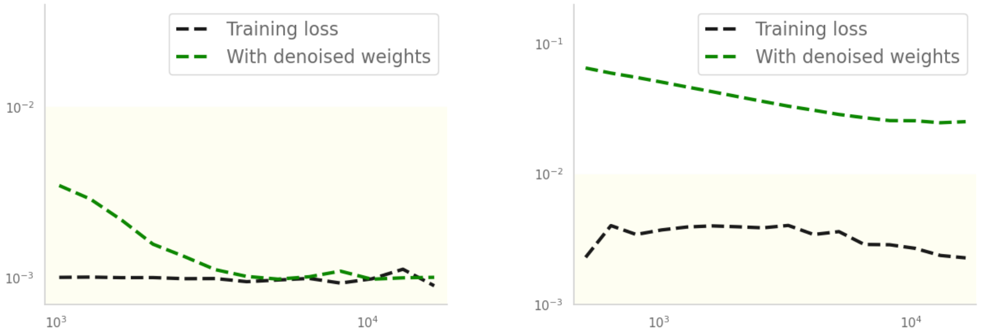

Figure 3.8: Loss value, as a function of L, in black for the trained weights $A_k^{(L)}$ and in green for the denoised weights $\widetilde{A}_k^{(L)} = L^{-\beta}\overline{A}_{k/L}$. Left: tanh activation and $\delta^{(L)} \in \mathbb{R}$. Right: ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$. Note 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.

α	$B = 8$	$B = 32$	$B = 128$
$\eta = .01$	$.69 \pm .02$	$.73 \pm .02$	$.67 \pm .02$
$\eta = .003$	$.59 \pm .05$	$.60 \pm .01$	$.58 \pm .01$
$\eta = .001$	$.58 \pm .01$	$.55 \pm .01$	$.53 \pm .01$

β	$B = 8$	$B = 32$	$B = 128$
$\eta = .01$	$.24 \pm .02$	$.29 \pm .05$	$.22 \pm .02$
$\eta = .003$	$.33 \pm .01$	$.41 \pm .06$	$.40 \pm .02$
$\eta = .001$	$.39 \pm .02$	$.43 \pm .02$	$.41 \pm .01$

We observe that the learning rate does affect α and β while keeping $\alpha + \beta$ around 1, and the batch size does not affect α or β. A plausible explanation for these observations is that a higher batch size means a more precise descent direction at the cost of efficiency, but the shape of the solution is not supposed to change.

3.3.3 Results for convolutional networks

We now consider the original ResNet with convolutional layers introduced in [73]. This architecture is close to the state-of-the-art methods used for image recognition tasks. We do not include batch normalization [80] since it only slightly improves the performance of the network while making the analysis significantly more complicated. The architecture is displayed in Figures 3.9 and 3.10.
Our network still possesses the skip connections from (3.1): the dynamics of the hidden state reads

$$h_{k+1} = \sigma\left(h_k + \Delta_k * \sigma\left(A_k * h_k\right) + F_k * h_k\right) \tag{3.9}$$

for $k = 0, \ldots, L-1$, where $\sigma = \text{ReLU}$. 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 (3.1). To lighten the notation, we omit the superscripts x (the input) and L (the number of layers).

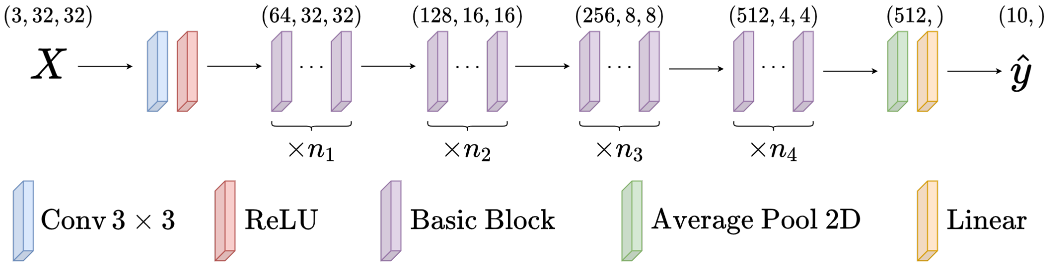

Figure 3.9: Residual architecture. There are 4 blocks that are respectively repeated n_1, n_2, n_3 and n_4 times. The network depth is $L = n_1 + n_2 + n_3 + n_4$. The Basic Block architecture is detailed in Figure 3.10.

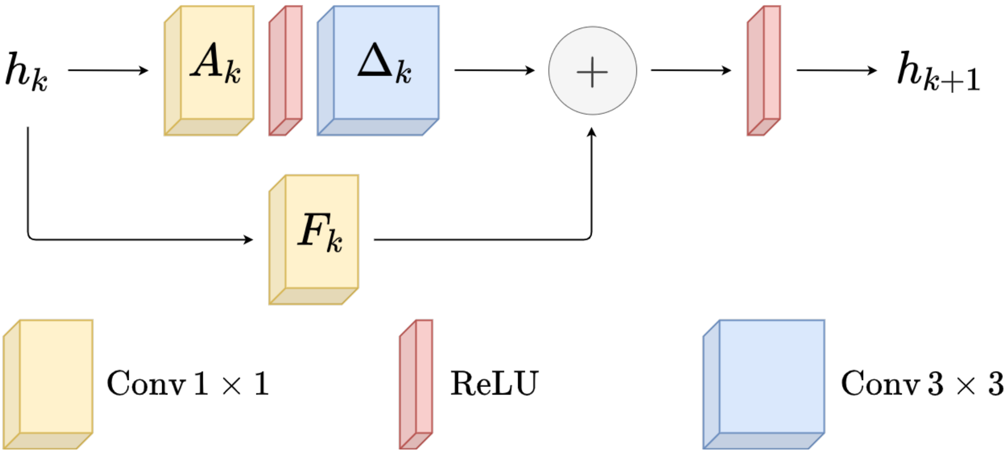

Figure 3.10: Basic Block from Figure 3.9. See (3.9) for details.

We train our residual networks at depths ranging from $L_{\min} = 8$ to $L_{\max} = 121$ on the CIFAR-10 [91] dataset with the unregularized relative entropy loss. Here, 'depth' is the number of residual connections. We note that a network with $L_{\max} = 121$ is already very deep. As a comparison, a standard ResNet-152 [73] has depth $L = 50$ in our framework.

Results. Table 3.3 shows the accuracy of our convolutional residual networks trained on an NVIDIA GeForce RTX 2080 GPU on the CIFAR-10 dataset. The results are in line with those of traditional ResNet architectures [73], even though our networks do not have batch normalization layers [80]. It is also noteworthy to add that our concept of depth is not that of traditional ResNets. We define the number of layers L as the number of skip connections in the network, that is the number of Δ_k kernels in (3.9).

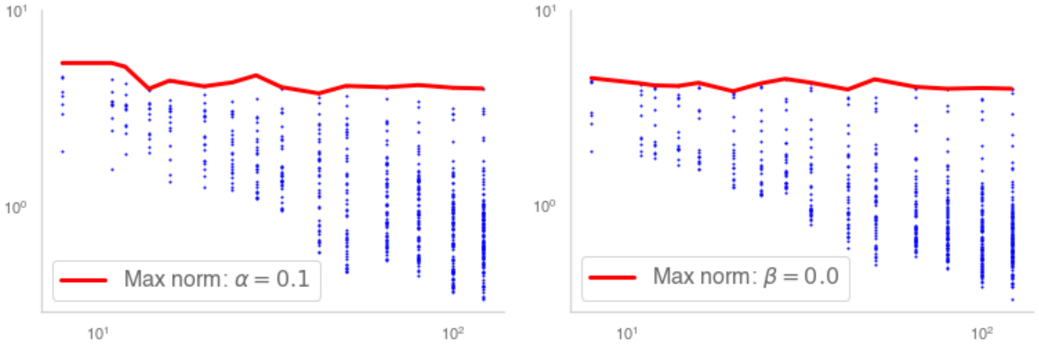

Figure 3.11: Scaling of $\Delta^{(L)}$ (left) and $A^{(L)}$ (right) against the network depth L for convolutional architectures on CIFAR-10. In blue: spectral norm of the kernels $\Delta_k^{(L)}$, resp. $A_k^{(L)}$, for $k = 0, \ldots, L-1$. In red: maximum norm, defined in Table 3.1. The plots are in log-log scale.

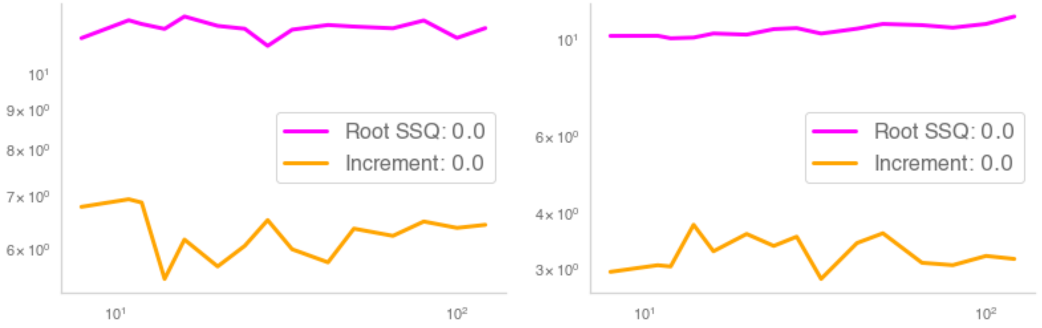

Figure 3.12: Scaling behavior of $\Delta^{(L)}$ (left) and $A^{(L)}$ (right). We plot in pink the root sum of squares and in orange the α-scaled norm of increments of $\Delta^{(L)}$ (left) and the β-scaled norm of increments of $A^{(L)}$ (right). Plots are in log-log scale. The root sum of squares and the scaled norm of increments are defined in Table 3.1. We obtain α and β from Figure 3.11.

Table 3.3: Learning error in % on CIFAR-10 for each network depth L.

L	8	11	12	14	16	20	24	28
Test error	6.64	6.37	6.32	5.98	6.25	5.98	6.24	7.03
L	33	42	50	65	80	100	121	
Test error	6.13	6.21	6.32	6.19	6.30	6.20	6.37	

As in Section 3.3.2, we investigate how the weights scale with depth and whether Scaling regime 1 or Scaling regime 2 holds true for convolutional layers. To that end, we follow the steps of [139] to get the singular values, and therefore the spectral norms, of the linear operators defined by the convolutional kernels $\Delta_k^{(L)}$ and $A_k^{(L)}$. Figure 3.11 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 $\beta = 0$.

We then use the values obtained for α and β to verify which Scaling regime holds. Figure 3.12 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 Scaling regime 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.

3.3.4 Summary: three scaling regimes

Our experiments show different scaling regimes for trained weights based on the network architecture.F or fully-connected layers with tanh activation and a shared $\delta^{(L)} \in \mathbb{R}$, we observe a behavior consistent with Scaling regime 1 for both the synthetic dataset and MNIST. For fully-connected layers with ReLU activation and $\delta_k^{(L)} \in \mathbb{R}$, we observe that Scaling regime 2 holds for the synthetic dataset and MNIST. We deduce that the results for fully-connected layers are consistent with our findings in Figure 3.1.

In the case of convolutional architectures trained on CIFAR-10 and presented in Section 3.3.3, 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 sparse scaling limits for both $\Delta^{(L)}$ and $A^{(L)}$. These results are consistent with previous evidence on the existence of sparse CNN representations for image recognition [109]. We stress that the setup for our CIFAR-10 experiments has been chosen to approach state-of-the-art performance with our generic architecture, as shown in Figures 3.9 and 3.10.

3.4 Deep network limit

In this section, we study the scaling limit of the hiddent state dynamics (3.1) under scaling regimes 1 and 2.

3.4.1 Scaling regime 1: ODE limit

First, we show that the scaling regime 1 together with a smooth and Lipschitz-continuous activation function lead to two ODE limits under different parameter regimes, including the neural ODE described in [28, 148, 67] as a special case.

We consider a setup which is consistent with Scaling regime 1 and $\delta^{(L)} = L^{-\alpha}$ for some $\alpha \geq 0$:

$$\begin{aligned} h_0^{(L)} &= x, \\ h_{k+1}^{(L)} &= h_k^{(L)} + L^{-\alpha}\,\sigma_d\left(A_k^{(L)} h_k^{(L)} + b_k^{(L)}\right), \end{aligned} \tag{3.10}$$

with

$$A_k^{(L)} = L^{-\beta}\overline{A}_{k/L}, \quad b_k^{(L)} = L^{-\beta}\overline{b}_{k/L}.$$

We focus our analysis on smooth activation functions.

Assumption 3.3 (Activation function)**.** *The activation function σ satisfies $\sigma \in \mathcal{C}^3(\mathbb{R}, \mathbb{R})$, $\sigma(0) = 0$, $\sigma'(0) = 1$, and has a bounded third derivative σ'''.*

Most smooth activation functions, including tanh, satisfy this condition. The boundedness of the third derivative σ''' may be relaxed to an exponential growth condition [124].

As observed in the numerical experiments, non-smooth activation functions such as ReLU lead to a different scaling regime to that of smooth functions.

We now describe ODE limits under Scaling regime 1. Let $\overline{H}^{(L)} : [0,1] \to \mathbb{R}^d$ be a continuous-time extension of the hidden states $h_k^{x,(L)}$:

$$\overline{H}_t^{(L)} := h_k^{x,(L)} \mathbb{1}_{\frac{k}{L} \leq t < \frac{k+1}{L}}, \quad k = 0, 1, \ldots, L. \tag{3.11}$$

Theorem 3.4 (ODE limits under Scaling regime 1)**.** *Under Assumption 3.3 and σ Lipschitz,*

- *Neural ODE limit [148, Lemma 4.6]: If $\alpha = 1$ and $\beta = 0$ and we further assume that $\overline{A} \in \mathcal{H}^1\big([0,1], \mathbb{R}^{d\times d}\big)$ and $\overline{b} \in \mathcal{H}^1\big([0,1], \mathbb{R}^d\big)$, then the interpolated hidden state dynamics (3.11) converge to the solution of the neural ODE*

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \sigma(\overline{A}_t H_t + \overline{b}_t), \qquad H_0 = x, \tag{3.12}$$

  in the sense that $\lim_{L\to\infty} \sup_{0\leq t\leq 1} \|H_t - \overline{H}_t\| = 0$.

- *A different ODE limit: If $\alpha + \beta = 1$ and $\beta > 0$, and there exist $M > 0$ and $\kappa > 0$ such that $\forall s, t \in [0,1]$, $\big\|\overline{A}_t - \overline{A}_s\big\| + \big\|\overline{b}_t - \overline{b}_s\big\| \leq M|t-s|^{\kappa/2}$, then the interpolated hidden state dynamics (3.11) converge to the solution of the following ODE*

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \overline{A}_t H_t + \overline{b}_t, \qquad H_0 = x, \tag{3.13}$$

  in the sense that $\lim_{L\to\infty} \sup_{0\leq t\leq 1} \|H_t - \overline{H}_t\| = 0$.

3.4.2 Scaling regime 2

Let $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ be a probability space with a $\mathbb{P}$-complete filtration $\mathbb{F} = (\mathcal{F}_t)_{t\geq 0}$. Let $(B_t^A)_{t\geq 0}$, resp. $(B_t^b)_{t\geq 0}$, be $d \times d$-dimensional, resp. d-dimensional, $\mathbb{F}$-standard uncorrelated Brownian motions. We consider a setup which is consistent with Section 3.3 where the noise part comes from the increment of some stochastic process and $\delta^{(L)} = L^{-\alpha}$ for some $\alpha \geq 0$:

$$\begin{aligned} h_0^{(L)} &= x, \\ h_{k+1}^{(L)} &= h_k^{(L)} + L^{-\alpha}\, \sigma_d \left(A_k^{(L)} h_k^{(L)} + b_k^{(L)}\right), \end{aligned} \tag{3.14}$$

with

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

where $(W_t^A)_{t\in[0,1]}$ and $(W_t^b)_{t\in[0,1]}$ are Itô processes [131] adapted to $\mathbb{F}$ and can be written in the form:

$$\begin{aligned} \left(\mathrm{d}W_t^A\right)_{ij} &= \left(U_t^A\right)_{ij} \mathrm{d}t + \sum_{k,l=1}^{d} \left(q_t^A\right)_{ijkl} \left(\mathrm{d}B_t^A\right)_{kl} \quad \text{for } i,j = 1,\ldots,d, \\ \mathrm{d}W_t^b &= U_t^b \mathrm{d}t + q_t^b\, \mathrm{d}B_t^b, \end{aligned} \tag{3.15}$$

with $W_0^A = 0$, $W_0^b = 0$, $q_t^A \in \mathbb{R}^{d,\otimes 4}$ and $q_t^b \in \mathbb{R}^{d\times d}$ for $t \in [0,1]$. We use the following notation for the quadratic variation of W^A and W^b:

$$\left[W^A\right]_t = \int_0^t \Sigma_u^A \,\mathrm{d}u, \quad \left[W^b\right]_t = \int_0^t \Sigma_u^b \,\mathrm{d}u, \tag{3.16}$$

where Σ^A and Σ^b are bounded processes with values respectively in $\mathbb{R}^{d,\otimes 4}$ and $\mathbb{R}^{d\times d}$. From (3.15) and (3.16), we have the quadratic variation process as follows:

$$\left(\Sigma_t^A\right)_{i_1 j_1 i_2 j_2} := \sum_{k,l=1}^{d} \left(q_t^A\right)_{i_1 j_1 kl} \left(q_t^A\right)_{i_2 j_2 kl}, \quad \text{for } i_1, j_1, i_2, j_2 = 1,\ldots,d, \quad \Sigma_t^b := q_t^b \left(q_t^b\right)^\top. \tag{3.17}$$

Here $(U_t^A)_{t\geq 0}$, $(U_t^b)_{t\geq 0}$, $(\Sigma_t^A)_{t\geq 0}$ and $(\Sigma_t^b)_{t\geq 0}$ are progressively measurable processes that satisfy the following conditions.

Assumption 3.5 (Regularity of the Ito processes (W^A, W^b) and continuous functions $(\overline{A}, \overline{b})$)**.** *We assume:*

(i) There exists a constant $C_1 > 0$ such that almost surely

$$\sup_{0\leq t\leq 1}\left\|U_t^A\right\| + \sup_{0\leq t\leq 1}\left\|U_t^b\right\| + \sup_{0\leq t\leq 1}\left\|\Sigma_t^A\right\| + \sup_{0\leq t\leq 1}\left\|\Sigma_t^b\right\| \leq C_1. \tag{3.18}$$

(ii) There exist $M > 0$ and $\kappa > 0$ such that $\forall s, t \in [0,1]$ almost surely

$$\left\|U_t^A - U_s^A\right\|^2 + \left\|U_t^b - U_s^b\right\|^2 + \left\|\Sigma_t^A - \Sigma_s^A\right\|^2 + \left\|\Sigma_t^b - \Sigma_s^b\right\|^2 \leq M|t-s|^\kappa, \tag{3.19}$$

and

$$\left\|\overline{A}_t - \overline{A}_s\right\|^2 + \left\|\overline{b}_t - \overline{b}_s\right\|^2 \leq M|t-s|^\kappa. \tag{3.20}$$

Note that (3.18) implies that $(U^A, U^B, \Sigma^A, \Sigma^B)$ are almost surely uniformly bounded and (3.19) implies that $(U^A, U^B, \Sigma^A, \Sigma^B)$ are almost surely Hölder continuous with exponent $\kappa/2$.

Lemma 3.6 (Uniform integrability)**.** *Under Assumption 3.5 (i), we have*

$$\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|W_s^A\right\|^{p_0}\right] \vee \mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|W_s^b\right\|^{p_0}\right] < \infty, \tag{3.21}$$

for any $p_0 > 1$.

Proof. By Minkowski's inequality and Assumption 3.5-(i),

$$\begin{aligned}
\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|W_s^A\right\|^{p_0}\right] &\leq 2^{p_0-1}\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\int_0^s U_t^A \mathrm{d}t\right\|^{p_0}\right] \\
&+ 2^{p_0-1}\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\left(\int_0^s \sum_{k,l=1}^d \left(q_t^A\right)_{ijkl}\left(\mathrm{d}B_t^A\right)_{kl}\right)_{i,j}\right\|^{p_0}\right] \\
&\leq 2^{p_0-1}C_1^{p_0} + 2^{p_0-1}\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\left(\int_0^s \sum_{k,l=1}^d \left(q_t^A\right)_{ijkl}\left(\mathrm{d}B_t^A\right)_{kl}\right)_{i,j}\right\|^{p_0}\right]
\end{aligned}$$

By the Burkholder-Davis-Gundy inequality and Assumption 3.5 (i),

$$\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\left(\int_0^s \sum_{k,l=1}^d \left(q_t^A\right)_{ijkl}\left(\mathrm{d}B_t^A\right)_{kl}\right)_{i,j}\right\|^{p_0}\right] \leq C_{p_0}\mathbb{E}\left[\left(\int_0^1 \Sigma_u^A \mathrm{d}u\right)^{p_0/2}\right] \leq C_{p_0}\, C_1^{p_0/2}.$$

Combining the two inequalities above, we get $\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|W_s^A\right\|^{p_0}\right] < \infty$. Similarly $\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|W_s^b\right\|^{p_0}\right] < \infty$ holds. □

Write $Q : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$, where each component Q_i is defined, for $i = 1, \dots, d$, as

$$Q_i(t,x) := \sum_{j,k=1}^{d} x_j x_k \left(\Sigma_t^A\right)_{ijik} + \Sigma_{t,ii}^b. \tag{3.22}$$

Let $\overline{H}^{(L)} : [0,1] \to \mathbb{R}^d$ be a continuous-time extension of the hidden states $h_k^{(L)}$:

$$\overline{H}_t^{(L)} := h_k^{x,(L)} \mathbb{1}_{\frac{k}{L} \leq t < \frac{k+1}{L}}, \quad k = 0, 1, \dots, L. \tag{3.23}$$

Assumption 3.7 (Uniform integrability). *There exist $p_1 > 4$ and a constant C_0 such that for all L,*

$$\mathbb{E}\left[\sup_{0 \leq t \leq 1} \left\| \overline{H}_t^{(L)} \right\|^{p_1}\right] \leq C_0. \tag{3.24}$$

Assumption 3.7 is standard in the convergence of approximation schemes for SDEs [75]. In practice, condition (3.24) is guaranteed throughout the training as both the inputs and the outputs of the network are bounded.

Let us now describe the intuition behind the deep network limit when $\beta > 0$. Denote $t_k = k/L$ and define for $s \in [t_k, t_{k+1})$:

$$\widetilde{M}_{k,s}^{(L)} := \left(W_s^A - W_{t_k}^A\right) h_k^{(L)} + \left(W_s^b - W_{t_k}^b\right) + L^{1-\beta} \overline{A}_{t_k} h_k^{(L)} (s - t_k) + L^{1-\beta} \overline{b}_{t_k} (s - t_k).$$

Using Itô's formula [81] to $\sigma\big(\widetilde{M}_{k,s}^{(L)}\big)$ for $s \in [t_k, t_{k+1})$, we obtain the following approximation

$$h_{k+1}^{(L)} - h_k^{(L)} = \delta^{(L)} \sigma\left(\widetilde{M}_{k,t_{k+1}}^{(L)}\right) \simeq D_1 + D_2 + D_3, \tag{3.25}$$

where

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

We observe from D_1 that (3.25) admits a diffusive limit only when $\alpha = 0$. In this case, we see that D_2 and D_3 do not explode only when $\beta \geq 1$, corresponding to a stochastic differential equation (SDE) limit that is diffusive. Another case where we obtain a non-trivial limit is when $\alpha > 0$ and $\alpha + \beta = 1$, which leads to an ODE limit.

We now provide a precise mathematical description of the different scaling limits of $\overline{H}^{(L)}$ for various values of α and β, using the concept of uniform convergence in L^2, also known as strong convergence. For a general exponent $p \geq 1$, we have the following definition.

Definition 3.8 (Uniform convergence in L^p). *Let $p \geq 1$ and $\mathcal{M}$ be the class of random functions $X : [0,1] \times \Omega \to \mathbb{R}^d$ such that*

$$\mathbb{E}\left[\sup_{t\in[0,1]} \|X(t)\|^p\right] < \infty.$$

We say that a sequence $(X^{(L)})_{L\in\mathbb{N}} \subset \mathcal{M}$ converges uniformly in L^p *to $X^* \in \mathcal{M}$ if*

$$\lim_{L\to\infty} \mathbb{E}\left[\sup_{0\leq t\leq 1} \left\|X_t^{(L)} - X_t^*\right\|^p\right] = 0. \tag{3.26}$$

We now show that Scaling regime 2 together with a smooth activation function lead to an ODE limit (which is different from the neural ODE) or a stochastic differential equation (SDE) depending on the values of α and β.

Theorem 3.9 (ODE limit under Scaling regime 2). *Under Assumptions 3.3, 3.5, and 3.7, if $\alpha > 0$, $\beta > 0$ and $\alpha + \beta = 1$, then the interpolated hidden state dynamics (3.23) converge uniformly in L^2 to the solution to the ODE*

$$\frac{\mathrm{d}H_t}{\mathrm{d}t} = \overline{A}_t H_t + \overline{b}_t, \qquad H_0 = x. \tag{3.27}$$

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 3.9, the limit (3.27) defines a linear input-output map behaving like a linear network [7]. This is different from the neural ODE (3.2), where the activation function σ appears in the limit.

Theorem 3.10 (SDE limit under Scaling regime 2). *Let Assumptions 3.3, 3.5 and 3.7 hold and let $\alpha = 0$ and $\beta \geq 1$. Denote H as the solution to the SDE*

$$\mathrm{d}H_t = \mathrm{d}W_t^A H_t + \mathrm{d}W_t^b + \frac{1}{2}\sigma''(0)Q(t,H_t)\,\mathrm{d}t + \mathbb{1}_{\beta=1}(\overline{A}_t H_t + \overline{b}_t)\,\mathrm{d}t, \tag{3.28}$$

with initial condition $H_0 = x$. If there exist $p_2 > 2$ such that $\mathbb{E}\left[\sup_{0\leq t\leq 1} \|H_t\|^{p_2}\right] < \infty$, then the interpolated hidden state dynamics (3.23) converge uniformly in L^2 to the solution of (3.28).

The proofs of Theorem 3.10 is given in Section 3.4.4. And the proof of Theorem 3.9 follows similar ideas. In particular, D_1 and D_2 vanish in the limit when $\alpha > 0$ in (3.25).

Interestingly, when the activation function σ is smooth, all limits in both Theorems 3.9 and 3.10 depend on the activation only through $\sigma'(0)$ (assumed to be 1 for simplicity) and $\sigma''(0)$. In contrast to the behavior of the neural ODE limit (3.2), the characteristics of σ away from 0 are not relevant to the limit. In addition, our proofs rely 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 3.10 is considered in [124], under the additional assumption that W^A and W^b are Brownian motions with constant drift. We consider a more general setup, where we introduce nonzero terms $\overline{A}$ and $\overline{b}$ and we allow W^A and W^b to be arbitrary Itô processes. Moreover, [124] prove weak convergence, which corresponds to convergence of quantities averaged across many trained networks with random independent initializations, whereas in practice, the training is done only once. Thus, the strong convergence, shown in Theorems 3.4 and 3.10, is a more relevant notion for studying the asymptotic behavior of deep neural networks.

Although the ResNet dynamics (3.14) is *not* expressed as an Euler scheme of a (ordinary or stochastic) differential equation, we nevertheless show strong convergence to a limitng ODE (in the case of Theorem 3.9) or SDE (in the case of Theorem 3.10), using techniques inspired by [75]. The challenge is to bound the difference between the ResNet dynamics and the Euler scheme of the limiting SDE. It is worth mentioning that the results in [75] hold for a class of time-homogeneous (Markov) diffusion processes whereas our result holds for Itô processes with bounded coefficients. This distinction is important for training neural networks since the "diffusion" assumption involves the distribution of the hidden state dynamics which can never be tested in practice. We can only verify the smoothness of the hidden state dynamics as detailed in Section 3.3. In addition, we also relax one technical condition assumed in [75], which is difficult to verify in practice. See Remark 3.12.

Note that we assume that the Ito processes W^A and W^b are driven by **uncorrelated** Brownian motions B^A and B^b. This assumption might look strong, but we pose it for ease of exposition: assuming a generic correlation structure between B^A and B^b would only a *cross-term* in the definition of Q.

3.4.3 Link with numerical experiments

Let us now discuss how the analysis above sheds light on the numerical results in Section 3.3.2 and Section 3.3.3.

Figure 3.2 shows that $\beta = 0.2$ and $\alpha = 0.7$ for the synthetic dataset with fully-connected layers and tanh activation function. This corresponds to the assumptions of Theorem 3.4 with the ODE limit (3.27). This is also consistent with the estimated decomposition in Figure 3.3 (right) where the noise part is negligible.

Regarding ReLU activation with fully-connected layers, we observe that $\beta + \alpha = 0.9$ from Figure 3.4 (left). Since ReLU is homogeneous of degree 1 (see Remark 3.2), $|\delta^{(L)}|$ can be moved inside σ, so without loss of generality we can assume $\alpha = 0$ and $\beta = 0.9$. If we replace the ReLU function by a smooth version σ^ϵ, then the limit is described by the stochastic differential equation (3.28). 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 and left for future work.

From the experiments with convolutional architectures, we observe that the maximum norm (Figure 3.11), the scaled norm of the increments, and the root sum of squares (Figure 3.12) are upper bounded as the number of layers L increases. This indicates that the weights fall into a sparse regime when L is large. In this case, there is no continuous ODE or SDE limit and Scaling regimes 1 and 2 both fail.

3.4.4 Detailed proofs

3.4.4.1 Proof of Theorem 3.4

It suffices to prove the second case with limit (3.13). First we show that there exists $C_\infty > 0$ such that

$$\sup_L \max_{0 \leq k \leq L} \left\| h_k^{(L)} \right\| \leq C_\infty. \tag{3.29}$$

Indeed, denote C_σ as the Lipschitz constant of σ. Then

$$\left\| h_{k+1}^{(L)} - h_k^{(L)} \right\| \leq \frac{C_\sigma}{L} \left\| \overline{A}_{t_k} h_k^{(L)} + \overline{b}_{t_k} \right\| \leq \frac{C_\sigma}{L} \left(A_{\max} + b_{\max} \right) \left(\left\| h_k^{(L)} \right\| + 1 \right),$$

where $A_{\max} := \sup_{0 \leq t \leq 1} \left\| \overline{A}_t \right\| < \infty$, $b_{\max} := \sup_{0 \leq t \leq 1} \left\| \overline{b}_t \right\| < \infty$, and $C_{\max} := A_{\max} + b_{\max}$. Hence

$$\left\| h_{k+1}^{(L)} \right\| \leq \left(\frac{C_\sigma C_{\max}}{L} + 1 \right) \left\| h_k^{(L)} \right\| + \frac{C_\sigma C_{\max}}{L}.$$

By induction:

$$\begin{aligned} \left\| h_j^{(L)} \right\| &\leq \|x\| \left(\frac{C_\sigma C_{\max}}{L} + 1 \right)^j + \frac{C_\sigma C_{\max}}{L} \sum_{i=1}^{j} \left(\frac{C_\sigma C_{\max}}{L} + 1 \right)^{i-1} \\ &\leq \left(\|x\| + C_\sigma C_{\max} \right) \left(\frac{C_\sigma C_{\max}}{L} + 1 \right)^L \\ &\to \left(\|x\| + C_\sigma C_{\max} \right) \exp\left(C_\sigma C_{\max} \right) \quad \text{as } L \to \infty. \end{aligned}$$

Hence (3.29) holds.

Denote $\Delta h_k^{(L)} := h_{k+1}^{(L)} - h_k^{(L)}$ and $M_k^{(L)}(h) := \overline{A}_{t_k} h + \overline{b}_{t_k}$. From (3.25) we have

$$\Delta h_k^{(L)} := h_{k+1}^{(L)} - h_k^{(L)} = L^{-\alpha} \sigma \left(L^{-\beta} M_k^{(L)} \left(h_k^{(L)} \right) \right).$$

Denote as well $\Delta h_{k,i}^{(L)}$ and $M_{k,i}^{(L)}$ the i-th element of $\Delta h_k^{(L)}$ and $M_k^{(L)}$, respectively. Applying a third-order Taylor expansion of σ around 0 with the help of Assumption 3.3, for $i = 1, 2, \dots, d$, we get

$$\begin{aligned}
\Delta h_{k,i}^{(L)} &= L^{-\alpha} \sigma_d \left(L^{-\beta} M_{k,i}^{(L)} \left(h_k^{(L)} \right) \right) \\
&= L^{-\beta-\alpha} M_{k,i}^{(L)} \left(h_k^{(L)} \right) + \frac{1}{2} \sigma''(0) L^{-2\beta-\alpha} \left(M_{k,i}^{(L)} \left(h_k^{(L)} \right) \right)^2 + \frac{1}{6} \sigma''' \left(\xi_{k,i}^{(L)} \right) L^{-3\beta-\alpha} \left(M_{k,i}^{(L)} \left(h_k^{(L)} \right) \right)^3 \\
&= L^{-1} M_{k,i}^{(L)} \left(h_k^{(L)} \right) + \frac{1}{2} \sigma''(0) L^{-\beta-1} \left(M_{k,i}^{(L)} (h_k^{(L)}) \right)^2 + \frac{1}{6} \sigma''' \left(\xi_{k,i}^{(L)} \right) L^{-2\beta-1} \left(M_{k,i}^{(L)} \left(h_k^{(L)} \right) \right)^3
\end{aligned} \tag{3.30}$$

with $\left| \xi_{k,i}^{(L)} \right| \le L^{-\beta} \left| \overline{A}_{t_k} h_k^{(L)} + \overline{b}_{t_k} \right|_i$. The last equation holds since $\alpha + \beta = 1$. Denote $t_k = k/L$ for $k = 0, 1, \dots, L$ as the uniform partition of the interval $[0, 1]$. For $t \in (t_k, t_{k+1}]$, define $\widetilde{H}_0^{(L)} := x = h_0^{(L)}$ and

$$\begin{aligned}
\widetilde{H}_t^{(L)} := h_k^{(L)} + (t - t_k) M_{k,i}^{(L)} \left(h_k^{(L)} \right) + \frac{1}{2} \sigma''(0) L^{-\beta-1} \left(M_{k,i}^{(L)} \left(h_k^{(L)} \right) \right)^2 \\
+ \frac{1}{6} \sigma''' \left(\xi_{k,i}^{(L)} \right) L^{-2\beta-1} \left(M_{k,i}^{(L)} \left(h_k^{(L)} \right) \right)^3.
\end{aligned}$$

Then we have $\widetilde{H}_{t_{k+1}}^{(L)} = h_k^{(L)} + \Delta h_k^{(L)} = h_{k+1}^{(L)}$ for all $k = 0, 1, \dots, L-1$. Recall $(H_t)_{t \in [0,1]}$ the solution to the ODE (3.13). Denote $d_k^{(L)}(t) := H_t - \widetilde{H}_t^{(L)}$ for $t \in (t_k, t_{k+1}]$ and define the errors

$$e_k^{(L),1} := \sup_{t_k < t \le t_{k+1}} \left\| \widetilde{H}_t^{(L)} - h_k^{(L)} \right\| \quad \text{and} \quad e_k^{(L),2} := \sup_{t_k < t \le t_{k+1}} \left\| d_k^{(L)}(t) \right\|.$$

We first bound $e_k^{(L),1}$. Note that by definition:

$$\begin{aligned}
e_k^{(L),1} &\le \left\| M_k^{(L),i} \left(h_k^{(L)} \right) \right\| L^{-1} + \frac{1}{2} \sigma''(0) L^{-\beta-1} \left(M_k^{(L),i} \left(h_k^{(L)} \right) \right)^2 + \frac{1}{6} c_0 L^{-2\beta-1} \left| M_k^{(L),i} \left(h_k^{(L)} \right) \right|^3 \\
&\le D_\infty L^{-1},
\end{aligned} \tag{3.31}$$

where $D_\infty := A_{\max} C_\infty + b_{\max} + \frac{1}{2} \sigma''(0) (A_{\max} C_\infty + b_{\max})^2 + \frac{1}{6} c_0 (A_{\max} C_\infty + b_{\max})^2$. Therefore we have

$$\lim_{L \to \infty} \sup_{0 \le k < L} e_k^{(L),1} = 0.$$

Next, we bound $e_k^{(L),2}$. For $t \in (t_{k+1}, t_{k+2}]$,

$$\begin{aligned} d_{k+1}^{(L)}(t) = d_k^{(L)}(t_{k+1}) - (t - t_{k+1})\, M_{k+1}^{(L),i}\left(h_{k+1}^{(L)}\right) + \int_{t_{k+1}}^{t} \left(\overline{A}_s H_s + \overline{b}_s\right) \mathrm{d}s \\ - \frac{1}{2}\sigma''(0) L^{-\beta-1} \left(M_{k+1}^{(L),i}\left(h_{k+1}^{(L)}\right)\right)^2 - \frac{1}{6}\sigma'''\left(\xi_{k+1}^{i}\right) L^{-2\beta-1} \left(M_{k+1}^{(L),i}\left(h_{k+1}^{(L)}\right)\right)^3 . \end{aligned} \tag{3.32}$$

Denote $c_0 := \sup_{x\in\mathbb{R}} |\sigma'''(x)| < \infty$, hence from (3.30) and (3.32),

$$\begin{aligned} e_{k+1}^{(L),2} \;\leq\; & e_k^{(L),2} + \sup_{t_{k+1} < t \leq t_{k+2}} \left\| \int_{t_{k+1}}^{t} \left(\left(\overline{A}_s H_s + \overline{b}_s\right) - \left(\overline{A}_{t_{k+1}} h_{k+1}^{(L)} + \overline{b}_{t_{k+1}}\right) \right) \mathrm{d}s \right\| \\ & + \frac{1}{2} \left|\sigma''(0)\right| L^{-\beta-1} \left\| M_{k+1}^{(L)}(h_{k+1}^{(L)}) \right\|^2 + \frac{1}{6} c_0 L^{-2\beta-1} \left\| M_{k+1}^{(L)}(h_{k+1}^{(L)}) \right\|^3 . \end{aligned}$$

Denote $H_{\max} := \sup_{0\leq t\leq 1} \|H_t\| < \infty$. Then,

$$\begin{aligned} \mathcal{E}_k^{(L)} \;:=\; & \sup_{t_{k+1} < t \leq t_{k+2}} \left\| \int_{t_{k+1}}^{t} \left(\left(\overline{A}_s H_s + \overline{b}_s\right) - \left(\overline{A}_{t_{k+1}} h_{k+1}^{(L)} + \overline{b}_{t_{k+1}}\right) \right) \mathrm{d}s \right\| \\ \leq\; & \sup_{t_{k+1} < t \leq t_{k+2}} \left\| \int_{t_{k+1}}^{t} \left(\overline{b}_s - \overline{b}_{t_{k+1}}\right) \mathrm{d}s \right\| + \sup_{t_{k+1} < t \leq t_{k+2}} \left\| \int_{t_{k+1}}^{t} \left(\overline{A}_s - \overline{A}_{t_{k+1}}\right) H_s \mathrm{d}s \right\| \\ & + \sup_{t_{k+1} < t \leq t_{k+2}} \left\| \overline{A}_{t_{k+1}} \int_{t_{k+1}}^{t} \left(H_s - h_{k+1}^{(L)}\right) \mathrm{d}s \right\| . \end{aligned}$$

Hence, we deduce

$$\begin{aligned} \mathcal{E}_k^{(L)} \;\leq\; & \int_{t_{k+1}}^{t_{k+2}} \left\|\overline{b}_s - \overline{b}_{t_{k+1}}\right\| \mathrm{d}s + H_{\max} \int_{t_{k+1}}^{t_{k+2}} \left\|\overline{A}_s - \overline{A}_{t_{k+1}}\right\| \mathrm{d}s + A_{\max} \int_{t_{k+1}}^{t_{k+2}} \left\| H_s - h_{k+1}^{(L)} \right\| \mathrm{d}s \\ \leq\; & M(1 + H_{\max}) \int_{t_{k+1}}^{t_{k+2}} |s - t_{t_{k+1}}|^{\kappa/2} \mathrm{d}s + A_{\max} \int_{t_{k+1}}^{t_{k+2}} \left\| H_s - h_{k+1}^{(L)} \right\| \mathrm{d}s \\ \leq\; & \frac{M}{1+\kappa/2} \left(1 + H_{\max}\right) L^{-(1+\kappa/2)} + \sup_{0\leq t\leq 1} \|\overline{A}_t\| L^{-1} \left(D_\infty L^{-1} + e_{k+1}^{(L),2} \right) . \end{aligned}$$

The last equation holds by (3.31). Then, we have for $L > A_{\max}$,

$$\begin{aligned} \left(1 - A_{\max} L^{-1}\right) e_{k+1}^{(L),2} \leq e_k^{(L),2} + \frac{M}{1+\kappa} \left(1 + H_{\max}\right) L^{-(1+\kappa)} + \frac{1}{2}\sigma''(0) L^{-(\beta+1)} \left(A_{\max} C_\infty + b_{\max}\right)^2 \\ + \frac{1}{6} c_0 L^{-(2\beta+1)} \left(A_{\max} C_\infty + b_{\max}\right)^3 + A_{\max} D_\infty L^{-2} \\ \leq e_k^{(L),2} + L^{-(1+\nu)} C_2, \end{aligned} \tag{3.33}$$

with $\nu := \min\{\kappa, \beta, 1\} > 0$ and C_2 a constant independent of k and L. Finally, when $L \geq G^{1/\gamma} + 2A_{\max}$ we have from (3.33):

$$e_0^{(L),2} \;\leq\; \frac{L^{-(1+\gamma)} G}{1 - A_{\max} L^{-1}} \leq \frac{1}{L - A_{\max}}, \tag{3.34}$$

and for $k = 0, \ldots, L-1$,

$$\begin{aligned} e_{k+1}^{(L),2} &\leq \frac{1}{1 - A_{\max} L^{-1}} \left(e_k^{(L),2} + L^{-(1+\gamma)} G \right) \\ &\leq \left(\frac{1}{1 - A_{\max} L^{-1}} \right)^{k+1} e_0^{(L),2} + L^{-(1+\gamma)} G \frac{\left(\frac{1}{1 - A_{\max} L^{-1}} \right)^{k+2} - 1}{\left(\frac{1}{1 - A_{\max} L^{-1}} \right) - 1} \\ &\leq \exp\left(2A_{\max} \frac{k+1}{L} \right) \frac{1}{L - A_{\max}} + L^{-\gamma} \frac{G}{A_{\max}} \exp\left(2A_{\max} \frac{k+2}{L} \right). \end{aligned} \quad (3.35)$$

(3.35) holds since $e_0^{(L),2} \leq \frac{1}{L - A_{\max}}$ and $\frac{1}{1 - A_{\max} L^{-1}} < 1 + 2A_{\max} L^{-1} \leq \exp(2A_{\max} L^{-1})$ when $L > 2A_{\max}$. Therefore, we conclude

$$\lim_{L \to \infty} \sup_{0 \leq k < L} e_k^{(L),2} = 0.$$

3.4.4.2 Proof of Theorem 3.9

We provide a complete proof of Theorem 3.9 for the case $\alpha = 0$ and $\beta = 1$. Other cases follow similarly. When $\alpha = 0$ and $\beta = 1$, we define the targeted SDE limit for the discrete scheme (3.14) as follows:

$$\mathrm{d}H_t = \mu(t, H_t)\mathrm{d}t + \mathrm{d}V_t^A\, H_t + \mathrm{d}V_t^b \quad \text{for } t \in [0,1]\,, \quad H_0 = x, \quad (3.36)$$

in which

$$\begin{aligned} \mu\,(t,h) &:= U_t^A\, h + U_t^b + \overline{A}_t h + \overline{b}_t + \frac{1}{2}\sigma''(0) Q(t,h), \\ \mathrm{d}V_t^A &:= \sum_{k,l=1}^{d} \left(q_t^A\right)_{ijkl} \left(\mathrm{d}B_t^A\right)_{kl}, \quad \mathrm{d}V_t^b := q_t^b\, \mathrm{d}B_t^b, \end{aligned} \quad (3.37)$$

with $V_0^A = 0$ and $V_0^b = 0$. Here the quadratic variation process $\frac{1}{2}\sigma''(0)Q(t,h)$ is the *Itô correction* term for the drift. On the one hand this correction term introduces non-linearity into the drift and makes the proof challenging. On the other hand, this term is the key for the convergence analysis. See (3.64) and (3.65).

Euler-Maruyama scheme of the limiting SDE. Denote $\Delta_L = 1/L$ as the sub-interval length and $t_k = k/L$, $k = 0, 1, \ldots, L$ as the uniform partition of the interval $[0,1]$. Further denote $\Delta V_k^A = V_{t_{k+1}}^A - V_{t_k}^A$ and $\Delta V_k^b = V_{t_{k+1}}^b - V_{t_k}^b$ as the increment of the stochastic processes. Define the Euler-Maruyama discretization scheme of the SDE (3.36) as:

$$\widehat{h}_{k+1}^{(L)} - \widehat{h}_k^{(L)} := \mu\left(t_k, \widehat{h}_k^{(L)}\right) \Delta_L + \Delta V_k^A\, \widehat{h}_k^{(L)} + \Delta V_k^b, \quad (3.38)$$

and the one-step forward increment follows:

$$f^{(L)}(k,h) := \mu\left(t_k,h\right)\Delta_L + \Delta V_k^A\, h + \Delta V_k^b. \tag{3.39}$$

Therefore (3.38) can be rewritten as $\widehat{h}_{k+1}^{(L)} = \widehat{h}_k^{(L)} + f^{(L)}\left(k, \widehat{h}_k^{(L)}\right)$.

Continuous-time extension. Recall that we extend the scheme $\left\{h_k^{(L)} : k = 0, \ldots, L\right\}$ to a continuous-time process $\overline{H}_t^{(L)}$ on $t \in [0,1]$ by a piecewise constant and right-continuous interpolation of $\{h_k^{(L)} : k = 0, \ldots, L-1\}$:

$$\overline{H}_t^{(L)} := \sum_{k=0}^{L} h_k^{(L)} \mathbf{1}_{t_k \le t < t_{k+1}}. \tag{3.40}$$

We call $\overline{H}_t^{(L)}$ the *continuous-time extension* (CTE) of $\{h_k^{(L)} : k = 0, \ldots, L-1\}$.

Continuous-time approximation. Denote

$$\begin{aligned} M_k^{(L)}(h) \;&:=\; \left(\mu\left(t_k,h\right) - \frac{1}{2}\sigma''(0)Q(t_k,h)\right)\Delta_L + \Delta V_k^A\, h + \Delta V_k^b \\ &=\; \left(U_{t_k}^A h + U_{t_k}^b + \overline{A}_{t_k} h + \overline{b}_{t_k}\right)\Delta_L + \Delta V_k^A\, h + \Delta V_k^b \\ &=:\; \widetilde{\mu}\left(t_k,h\right)\Delta_L + \Delta V_k^A\, h + \Delta V_k^b, \end{aligned} \tag{3.41}$$

and from (3.25) we thus have

$$\Delta h_k^{(L)} := h_{k+1}^{(L)} - h_k^{(L)} = \sigma\left(M_k^{(L)}\left(h_k^{(L)}\right)\right).$$

Denote $\Delta h_{k,i}^{(L)}$ and $M_{k,i}^{(L)}$ the i-th element of $\Delta h_k^{(L)}$ and $M_k^{(L)}$, respectively. Applying a third-order Taylor expansion of σ around 0 with the help of Assumption 3.3, for $i = 1, 2, \ldots, d$, we get

$$\begin{aligned} &\Delta h_{k,i}^{(L)} = \sigma\left(M_{k,i}^{(L)}\left(h_k^{(L)}\right)\right) \\ &= M_{k,i}^{(L)}\left(h_k^{(L)}\right) + \frac{1}{2}\sigma''(0) M_{k,i}^{(L)}\left(h_k^{(L)}\right)^2 + \frac{1}{6}\sigma'''\left(\xi_{k,i}^{(L)}\right) M_{k,i}^{(L)}\left(h_k^{(L)}\right)^3 \\ &= \underbrace{\mu_i\left(t_k, h_k^{(L)}\right)\Delta_L + \left(\Delta V_k^A\, h_k^{(L)}\right)_i + \left(\Delta V_k^b\right)_i}_{f_i^{(L)}\left(k, h_k^{(L)}\right)} + \underbrace{\frac{1}{2}\sigma''(0)\left(M_{k,i}^{(L)}\left(h_k^{(L)}\right)^2 - Q_i\left(t_k, h_k^{(L)}\right)\right)}_{N_{k,i}^{(L)}\left(h_k^{(L)}\right)} \\ &\quad + \frac{1}{6}\sigma'''\left(\xi_{k,i}^{(L)}\right) M_{k,i}^{(L)}\left(h_k^{(L)}\right)^3 \\ &= f_i^{(L)}\left(k, h_k^{(L)}\right) + N_{k,i}^{(L)}\left(h_k^{(L)}\right) + \frac{1}{6}\sigma'''\left(\xi_{k,i}^{(L)}\right) M_{k,i}^{(L)}\left(h_k^{(L)}\right)^3, \end{aligned}$$

with $\left|\xi_{k,i}^{(L)}\right| < \left|M_{k,i}^{(L)}\left(h_k^{(L)}\right)\right|$. The increment of the ResNet $\Delta h_{k,i}^{(L)}$ has two parts: the increment of the Euler-Maruyama scheme $f_i^{(L)}\left(k, h_k^{(L)}\right)$ and the residual

$$D_{k,i}^{(L)}\left(h_k^{(L)}\right) := \frac{1}{6}\sigma'''\left(\xi_{k,i}^{(L)}\right) M_{k,i}^{(L)}\left(h_k^{(L)}\right)^3 + N_{k,i}^{(L)}\left(h_k^{(L)}\right). \tag{3.42}$$

It is clear from here that the Euler-Maruyama scheme of the limiting SDE is different from the ResNet dynamics. Hence classical results on the convergence of discrete SDE schemes cannot be applied directly.

In our analysis it will be more natural to work with the following *continuous-time approximation* (CTA), defined as

$$\widetilde{H}_t^{(L)} := h_0^{(L)} + \int_0^t \mu\left(t_{k_s}, \overline{H}_s^{(L)}\right) \mathrm{d}s + \int_0^t \left(\mathrm{d}V_s^A \, \overline{H}_s^{(L)} + \mathrm{d}V_s^b\right) + \sum_{k<Lt} D_k^{(L)}\left(h_k^{(L)}\right), \tag{3.43}$$

where $D_k^{(L)}(h) = \left(D_{k,1}^{(L)}(h), \ldots, D_{k,d}^{(L)}(h)\right)^\top$ and k_s is the integer for which $s \in [t_{k_s}, t_{k_s+1})$ for a given $s \in [0,1)$.

Here $\widetilde{H}_t^{(L)}$ approximates the CTE (3.40) with a continuous version, with interpolations both in time and in space, of the $f^{(L)}(k,h)$ part while the residual term $D_k^{(L)}(h)$ remains the same. By design we have $\widetilde{H}_{t_k}^{(L)} = \overline{H}_{t_k}^{L} = h_k^{(L)}$, that is, $\widetilde{H}_t^{(L)}$ and $\overline{H}_t^{(L)}$ coincide with the discrete solution at grid points t_k, $k = 0, 1, \ldots, L-1$. This relationship is instrumental in order to control the error.

We will first study the difference between $\widetilde{H}$ and $h^{(L)}$, and then the difference between $\overline{H}$ and $h^{(L)}$, in the supremum norm. The sum of the two will give a bound for the error of the discrete approximation.

3.4.4.3 Preliminary result

Lemma 3.11 (Local Lipschitz condition and uniform integrability)**.** *Under the assumptions from Theorem 3.10, we have the folllowing results:*

(i) For each $R > 0$, there exists a constant C_R, depending only on R, such that almost surely we have

$$\|\mu(t,x) - \mu(t,y)\|^2 \leq C_R \|x-y\|^2, \ \forall t \in [0,1] \ \forall x,y \in \mathbb{R}^d \text{ with } \|x\| \vee \|y\| \leq R, \tag{3.44}$$

where μ is defined in (3.37).

(ii) There exist some constants $p > 2$ and $C > 0$ such that

$$\mathbb{E}\left[\sup_{0\leq t\leq 1} \left\|\widetilde{H}_t^{(L)}\right\|^p\right] \vee \mathbb{E}\left[\sup_{0\leq t\leq 1} \|H_t\|^p\right] \leq C. \tag{3.45}$$

Remark 3.12. *Note that [75] assumes the uniform integrability condition for $\widetilde{H}_t^{(L)}$ which is difficult to verify in practice. Here we relax this condition by only assuming the uniform integrability condition for the ResNet dynamics $\{h_k^{(L)} : k = 0, \ldots, L\}$, see Assumption 3.7. We can then prove (3.45) under Assumption 3.7 and some properties of the Itô processes.*

Proof of Lemma 3.11. First, there exists $C_2 > 0$ such that

$$\|Q(t,x) - Q(t,y)\| \leq C_2 \|x-y\| \; \|x+y\| \leq 2C_2 R \|x-y\|, \tag{3.46}$$

since $Q(t,x)$ is quadratic in x and $\sup_{0\leq t\leq 1} \|\Sigma_t^A\| \leq C_1$. Then,

$$\begin{aligned} \|\mu(t,x) - \mu(t,y)\|^2 &= \left\| U_t^A (x-y) + \overline{A}_t (x-y) + \frac{1}{2}\sigma''(0) \left(Q(t,x) - Q(t,y)\right)\right\|^2 \\ &\leq \left(3 \max_{t\in[0,1]} \left\|U_t^A\right\| + 3 \max_{t\in[0,1]} \left\|\overline{A}_t\right\| + 3\left|\sigma''(0)\right| C_2^2 R^2\right) \|x-y\|^2. \end{aligned}$$

Note that $\max_{t\in[0,1]} \left\|\overline{A}_t\right\| < \infty$ since $\overline{A} \in \mathcal{C}^0\left([0,1], \mathbb{R}^d\right)$ and $\max_{t\in[0,1]} \left\|U_t^A\right\| < C_1$ almost surely according to (3.18), respectively. Therefore (3.44) holds by taking $C_R = 3\max_{t\in[0,1]} \left\|U_t^A\right\| + 3\max_{t\in[0,1]} \left\|\overline{A}_t\right\| + 3\left|\sigma''(0)\right| C_2^2 R^2$.

Thanks to the assumption in Theorem 3.10, there exists a constant $C_3 > 0$ such that $\mathbb{E}\left[\sup_{0\leq t\leq 1} \|H_t\|^{p_1}\right] \leq C_3$, then we only need to show that (3.45) holds for $\widetilde{H}$ for some $p > 2$. To see this, let k_s be the integer for which $s \in [t_{k_s}, t_{k_s+1})$ for a given $s \in [0,1)$. Then

$$\begin{aligned} \overline{H}_s^{(L)} - \widetilde{H}_s^{(L)} &= h_{k_s}^{(L)} - \left(h_{k_s}^{(L)} + \int_{t_{k_s}}^s \mu\left(t_{k_r}, \overline{H}_r^{(L)}\right) \mathrm{d}r + \int_{t_{k_s}}^s \left(\mathrm{d}V_r^A \overline{H}_r^{(L)} + \mathrm{d}V_r^b\right)\right) \\ &= -\mu\left(t_{k_s}, h_{k_s}^{(L)}\right)(s - t_{k_s}) - \left(V_s^A - V_{t_{k_s}}^A\right) h_{k_s}^{(L)} - \left(V_s^b - V_{t_{k_s}}^b\right). \end{aligned}$$

Hence, by the Minkowski inequality,

$$\begin{aligned} \left\|\overline{H}_s^{(L)} - \widetilde{H}_s^{(L)}\right\|^p &\leq 3^{p-1}\left(\left\|\mu\left(t_{k_s}, h_{k_s}^{(L)}\right)\right\|^p (\Delta_L)^p + \left\|h_{k_s}^{(L)}\right\|^p \left\|V_s^A - V_{t_{k_s}}^A\right\|^p + \left\|V_s^b - V_{t_{k_s}}^b\right\|^p\right) \\ &\leq C_4\left(\left\|h_{k_s}^{(L)}\right\|^{2p} + \left\|h_{k_s}^{(L)}\right\|^p + 1 + \left\|h_{k_s}^{(L)}\right\|^p \left\|V_s^A - V_{t_{k_s}}^A\right\|^p + \left\|V_s^b - V_{t_{k_s}}^b\right\|^p\right) \end{aligned} \tag{3.47}$$

for some $C_4 > 0$, as $\mu(t,h)$ is quadratic in h. The value of $p > 2$ will be determined later. From (3.47), we get

$$\begin{aligned} &\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)} - \widetilde{H}_s^{(L)}\right\|^p\right] \\ \leq\; & C_4\left(\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)}\right\|^{2p}\right] + \mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)}\right\|^{p}\right] + 1\right) \\ &+ C_5\left(\left(\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)}\right\|^{2p}\right]\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|V_s^A - V_{t_{k_s}}^A\right\|^{2p}\right]\right)^{1/2} + \mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|V_s^b - V_{t_{k_s}}^b\right\|^{p}\right]\right) \\ \leq\; & C_4\left(\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)}\right\|^{2p}\right] + \mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)}\right\|^{p}\right] + 1\right) \\ &+ C_6\left(\left(\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\overline{H}_s^{(L)}\right\|^{2p}\right]\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|V_s^A\right\|^{2p}\right]\right)^{1/2} + \mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|V_s^b\right\|^{p}\right]\right), \end{aligned} \tag{3.48}$$

for some constants $C_4, C_5, C_6 > 0$ independent of L, R and δ. The first inequality holds by the Hölder and (3.48) holds by the Minkowski inequality. Take $p = \frac{1}{2}p_1 > 2$. Then, (3.48) is bounded thanks to Assumption 3.7 for $\mathbb{E}\left[\sup_{0\le t\le 1}\|\overline{H}_t^{(L)}\|^{2p}\right] < \infty$, and we have $\mathbb{E}\left[\sup_{0\le s\le 1}\left\|W_s^A\right\|^p\right] < \infty$ and $\mathbb{E}\left[\sup_{0\le s\le 1}\left\|W_s^b\right\|^p\right] < \infty$ by (3.21). Hence, by the Minkowski inequality, we have

$$\mathbb{E}\left[\sup_{0\le t\le 1}\left\|\widetilde{H}_t^{(L)}\right\|^p\right] \le 2^{p-1}\mathbb{E}\left[\sup_{0\le s\le 1}\left\|\overline{H}_s^{(L)} - \widetilde{H}_s^{(L)}\right\|^p\right] + 2^{p-1}\mathbb{E}\left[\sup_{0\le t\le 1}\left\|\overline{H}_t^{(L)}\right\|^p\right] < \infty.$$

□

3.4.4.4 Proof of Theorem 3.10

We are now ready to show the proof of Theorem 3.10.

Proof. Let us define two stopping times to utilize the local Lipschitz property of μ:

$$\tau_R := \inf\left\{t \ge 0 : \left\|\widetilde{H}_t^{(L)}\right\| \ge R\right\}, \quad \rho_R := \inf\left\{t \ge 0 : \|H_t\| \ge R\right\}, \quad \theta_R := \tau_R \wedge \rho_R, \tag{3.49}$$

and define the approximation errors

$$e_1(t) := \widetilde{H}_t^{(L)} - H_t, \quad \text{and} \quad e_2(t) := \widetilde{H}_t^{(L)} - \overline{H}_t^{(L)}. \tag{3.50}$$

The proof contains two steps. The first step is to show $\lim_{L\to\infty}\mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\right] = 0$ and the second step is to show $\lim_{L\to\infty}\mathbb{E}\left[\sup_{0\le t\le 1}\|e_2(t)\|^2\right] = 0$.
Following the idea in [75], we first show that for any $\delta > 0$ (to be determined later):

$$\mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\right] \le \mathbb{E}\left[\sup_{0\le t\le 1}\left\|\widetilde{H}_{t\wedge\theta_R}^{(L)} - H_{t\wedge\theta_R}\right\|^2\right] + \frac{2^{p+1}\delta C}{p} + \frac{2(p-2)C}{p\delta^{2/(p-2)}R^p}, \tag{3.51}$$

where C and p are defined in (3.45). To see this, recall that by Young's inequality, for $r^{-1} + q^{-1} = 1$, we have

$$ab = \delta^{1/r}a \cdot \delta^{1/q-1}b \le \frac{\delta}{r}a^r + \frac{1}{q\delta^{q/r}}b^q, \quad \forall a, b, \delta > 0. \tag{3.52}$$

First decompose the left-hand side of (3.51) to obtain, for all $\delta > 0$,

$$\begin{aligned}\mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\right] &= \mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\mathbb{1}_{\{\tau_R>1,\rho_R>1\}}\right] + \mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\mathbb{1}_{\{\tau_R\le 1 \text{ or } \rho_R\le 1\}}\right] \\ &\le \mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t\wedge\theta_R)\|^2\mathbb{1}_{\{\theta_R>1\}}\right] + \frac{2\delta}{p}\mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^p\right] \\ &\quad + \frac{1-2/p}{\delta^{2/(p-2)}}\mathbb{P}\left(\tau_R \le 1 \text{ or } \rho_R \le 1\right).\end{aligned} \tag{3.53}$$

where we apply (3.52) with $r = p/2$ to the second term. Now

$$\mathbb{P}(\tau_R \leq 1) = \mathbb{E}\left[\mathbb{1}_{\{\tau_R \leq 1\}} \frac{\|\widetilde{H}^{(L)}_{\tau_R}\|^p}{R^p}\right] \leq \frac{1}{R^p}\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|\widetilde{H}^{(L)}_t\right\|^p\right] \leq \frac{C}{R^p}. \tag{3.54}$$

A similar result can be derived for ρ_R, so that we have

$$\mathbb{P}(\tau_R \leq 1 \text{ or } \rho_R \leq 1) \leq \mathbb{P}(\tau_R \leq 1) + \mathbb{P}(\rho_R \leq 1) \leq \frac{2C}{R^p}. \tag{3.55}$$

Using the inequalities in (3.54)–(3.55), along with

$$\mathbb{E}\left[\sup_{0\leq t\leq 1}\|e_1(t)\|^p\right] \leq 2^{p-1}\mathbb{E}\left[\sup_{0\leq t\leq 1}\left(\left\|\widetilde{H}^{(L)}_t\right\|^p + \|H_t\|^p\right)\right] \leq 2^p\, C \tag{3.56}$$

in (3.53), we show the desired result (3.51).

To obtain a uniform bound on $\widetilde{H} - H$, we bound the first term on the right-hand side of (3.51). Using the definition of the targeted SDE limit in (3.36):

$$H_{t\wedge\theta_R} := H_0 + \int_0^{t\wedge\theta_R}\mu(s, H_s)\mathrm{d}s + \int_0^{t\wedge\theta_R}\left(\mathrm{d}W^A_s\, H_s + \mathrm{d}W^b_s\right),$$

and the continuous-time approximation (3.43), we get

$$\begin{aligned}
\left\|\widetilde{H}^{(L)}_{t\wedge\theta_R} - H_{t\wedge\theta_R}\right\|^2 &= \Bigg\|\int_0^{t\wedge\theta_R}\left(\mu\left(t_{k_s}, \overline{H}^{(L)}_s\right)\mathrm{d}s - \mu\left(s, H_s\right)\right)\mathrm{d}s \\
&\qquad + \int_0^{t\wedge\theta_R}\mathrm{d}W^A_s\;\left(\overline{H}^{(L)}_s - H_s\right) + \sum_{k<L(t\wedge\theta_R)} D^{(L)}_k\left(h^{(L)}_k\right)\Bigg\|^2 \\
&= \Bigg\|\int_0^{t\wedge\theta_R}\left(\mu\left(s, \overline{H}^{(L)}_s\right) - \mu(s, H_s) + \mu\left(t_{k_s}, \overline{H}^{(L)}_s\right) - \mu\left(s, \overline{H}^{(L)}_s\right)\right)\mathrm{d}s \\
&\qquad + \int_0^{t\wedge\theta_R}\mathrm{d}W^A_s\;\left(\overline{H}^{(L)}_s - H_s\right) + \sum_{k<L(t\wedge\theta_R)} D^{(L)}_k\left(h^{(L)}_k\right)\Bigg\|^2.
\end{aligned}$$

We first bound the above using Cauchy-Schwarz inequality:

$$\begin{aligned}
&\left\|\widetilde{H}^{(L)}_{t\wedge\theta_R} - H_{t\wedge\theta_R}\right\|^2 \\
\leq\;& 4\left[\int_0^{t\wedge\theta_R}\left\|\mu\left(s, \overline{H}^{(L)}_s\right)\mathrm{d}s - \mu\left(s, H_s\right)\right\|^2\mathrm{d}s\right] + 4\left\|\int_0^{t\wedge\theta_R}\mathrm{d}W^A_s\;\left(\overline{H}^{(L)}_s - H_s\right)\right\|^2 \\
&+ 4\left[\int_0^{t\wedge\theta_R}\left\|\mu\left(t_{k_s}, \overline{H}^{(L)}_s\right)\mathrm{d}s - \mu\left(s, \overline{H}^{(L)}_s\right)\right\|^2\mathrm{d}s\right] + 4\left\|\sum_{k<L(t\wedge\theta_R)} D^{(L)}_k\left(h^{(L)}_k\right)\right\|^2.
\end{aligned}$$

Now, from the local Lipschitz condition (3.44) and Doob's martingale inequality [131], we have for any $\tau \leq 1$,

$$
\begin{aligned}
&\mathbb{E}\left[\sup_{0\leq t\leq\tau}\left\|\widetilde{H}^{(L)}_{t\wedge\theta_R}-H_{t\wedge\theta_R}\right\|^2\right]\\
&\leq 4\left(C_R+4C_1^2\right)\mathbb{E}\int_0^{\tau\wedge\theta_R}\left\|\overline{H}^{(L)}_s-H_s\right\|^2\mathrm{d}s\\
&\quad+4\,\mathbb{E}\left[\int_0^{t\wedge\theta_R}\left\|\mu\left(t_{k_s},\overline{H}^{(L)}_s\right)\mathrm{d}s-\mu\left(s,\overline{H}^{(L)}_s\right)\right\|^2\mathrm{d}s\right]+4\sum_{k\leq L\tau}\mathbb{E}\left\|D^{(L)}_k\left(h^{(L)}_k\right)\mathbb{1}_{\|h^{(L)}_k\|\leq R}\right\|^2\\
&\leq C'_R\int_0^{\tau}\mathbb{E}\left[\sup_{0\leq r\leq s}\left\|\widetilde{H}^{(L)}_{r\wedge\theta_R}-H_{r\wedge\theta_R}\right\|^2\right]\mathrm{d}s+C'_R\underbrace{\mathbb{E}\int_0^{\tau\wedge\theta_R}\left\|\overline{H}^{(L)}_s-\widetilde{H}^{(L)}_s\right\|^2\mathrm{d}s}_{\textcircled{1}}\\
&\quad+4\underbrace{\mathbb{E}\left[\int_0^{t\wedge\theta_R}\left\|\mu\left(t_{k_s},\overline{H}^{(L)}_s\right)-\mu\left(s,\overline{H}^{(L)}_s\right)\right\|^2\mathrm{d}s\right]}_{\textcircled{2}}+4\underbrace{\mathbb{E}\left[\sup_{0\leq t\leq\tau}\left\|\sum_{k<L(t\wedge\theta_R)}D^{(L)}_k\left(h^{(L)}_k\right)\right\|^2\right]}_{\textcircled{3}}
\end{aligned}
\tag{3.57}
$$

where $C'_R := 8\left(C_R+4C_1^2\right)$. First, we give an upper bound for $\textcircled{2}$. By the Cauchy–Schwarz inequality,

$$
\begin{aligned}
\|\mu(t,h)-\mu(s,h)\|^2\leq 5\,\Big(&\left\|U^A_t-U^A_s\right\|^2\|h\|^2+\left\|U^b_t-U^b_s\right\|^2+\left\|\overline{A}_s-\overline{A}_t\right\|\|h\|^2+\left\|\bar{b}_t-\bar{b}_s\right\|^2\\
&+\frac{1}{2}\sigma''(0)\,\|Q(t,h)-Q(s,h)\|^2\Big).
\end{aligned}
$$

Hence, for $h\in\mathbb{R}^d$, the following holds almost surely by (3.19):

$$
\|\mu(t,h)-\mu(s,h)\|^2\leq C_M\,|t-s|^{\kappa}\left(1+\|h\|^2+\|h\|^4\right). \tag{3.58}
$$

Under Assumption 3.7, there exists a constant $\widetilde{C}_0>0$ such that

$$
\mathbb{E}\left[\sup_{0\leq t\leq 1}\left(\left\|\overline{H}^{(L)}_t\right\|^4+\left\|\overline{H}^{(L)}_t\right\|^2\right)\right]\leq\widetilde{C}_0.
$$

Hence by Tonelli's theorem,

$$
\begin{aligned}
\mathbb{E}\left[\int_0^{t\wedge\theta_R}\left\|\mu\left(t_{k_s},\overline{H}^{(L)}_s\right)-\mu\left(s,\overline{H}^{(L)}_s\right)\right\|^2\mathrm{d}s\right]&\leq\int_0^1\mathbb{E}\left[\left\|\mu\left(t_{k_s},\overline{H}^{(L)}_s\right)-\mu\left(s,\overline{H}^{(L)}_s\right)\right\|^2\right]\mathrm{d}s\\
&\leq(\widetilde{C}_0+1)C_M L\left(\int_0^{1/L}r^{\kappa}\mathrm{d}r\right)\\
&=\frac{(\widetilde{C}_0+1)C_M}{1+\kappa}L^{-\kappa}.
\end{aligned}
\tag{3.59}
$$

Upper bound on ③. Define the following discrete filtration

$$\mathcal{G}_k := \sigma\Big(U^A_s, U^A_s, q^A_s, q^b_s, B^A_s, B^b_s \;:\; s \le t_{k+1}\Big). \tag{3.60}$$

Note that $h^{(L)}_k$ is $\mathcal{G}_{k-1}$-measurable but not $\mathcal{G}_k$-measurable. Define for $k = 0, \dots, L-1$ and for $i = 1, \dots, d$:

$$\begin{aligned}
X^{(L)}_{k,i} &:= \left(\left(\Delta V^A_k\, h^{(L)}_k\right)_i + \left(\Delta V^b_k\right)_i\right)^2 - \mathbb{E}\left[\left(\left(\Delta V^A_k\, h^{(L)}_k\right)_i + \left(\Delta V^b_k\right)_i\right)^2 \middle| \mathcal{G}_{k-1}\right]\\
Y^{(L)}_{k,i} &:= \mathbb{E}\left[\left(\left(\Delta V^A_k\, h^{(L)}_k\right)_i + \left(\Delta V^b_k\right)_i\right)^2 \middle| \mathcal{G}_{k-1}\right] - Q_i\left(t_k, h^{(L)}_k\right)\Delta_L\\
J^{(L)}_{k,i} &:= \widetilde{\mu}_i(t,h)^2(\Delta_L)^2 + 2\widetilde{\mu}_i(t,h)\Delta_L\left(\left(\Delta V^A_k\, h\right)_i + \left(\Delta V^b_k\right)_i\right).
\end{aligned}$$

We can then decompose

$$\begin{aligned}
D^{(L)}_{k,i}\left(h^{(L)}_k\right) &= \frac{1}{6}\sigma'''\left(\xi^{(L)}_{k,i}\right) M^{(L)}_{k,i}\left(h^{(L)}_k\right)^3 + N^{(L)}_{k,i}\left(h^{(L)}_k\right)\\
&= \frac{1}{6}\sigma'''\left(\xi^{(L)}_{k,i}\right) M^{(L)}_{k,i}\left(h^{(L)}_k\right)^3 + \frac{1}{2}\sigma''(0)\left(X^{(L)}_{k,i} + Y^{(L)}_{k,i} + J^{(L)}_{k,i}\right).
\end{aligned}$$

Hence, we deduce the following bound on ③ by Cauchy-Schwarz.

$$\begin{aligned}
&\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k<L(t\wedge\theta_R)} D^{(L)}_{k,i}\left(h^{(L)}_k\right)\right|^2\right]\\
&\le \sigma''(0)^2\,\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k<L(t\wedge\theta_R)} X^{(L)}_{k,i}\right|^2 + \left|\sum_{k<L(t\wedge\theta_R)} Y^{(L)}_{k,i}\right|^2 + L\sum_{k<L(t\wedge\theta_R)}\left|J^{(L)}_{k,i}\right|^2\right]\\
&\quad + 4\,\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k<L(t\wedge\theta_R)}\frac{1}{6}\sigma'''\left(\xi^{(L)}_{k,i}\right) M^{(L)}_{k,i}\left(h^{(L)}_k\right)^3\right|^2\right]\\
&\le \sigma''(0)^2\,\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k<Lt} X^{(L)}_{k,i}\mathbb{1}_{\|h^{(L)}_k\|\le R}\right|^2\right] + \sigma''(0)^2\,\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k<Lt} Y^{(L)}_{k,i}\mathbb{1}_{\|h^{(L)}_k\|\le R}\right|^2\right]\\
&\quad + \sigma''(0)^2 L\sum_{k=0}^{L-1}\mathbb{E}\left[\left|J^{(L)}_{k,i}\right|^2\mathbb{1}_{\|h^{(L)}_k\|\le R}\right] + \frac{1}{9}\sigma'''\left(\xi^{(L)}_{k,i}\right)^2 L\sum_{k=0}^{L-1}\mathbb{E}\left[M^{(L)}_{k,i}\left(h^{(L)}_k\right)^6\mathbb{1}_{\|h^{(L)}_k\|\le R}\right].
\end{aligned} \tag{3.61}$$

We provide an upper bound for each of the four terms in (3.61). For the first term, denote $\widetilde{X}^{(L)}_{k,i} := X^{(L)}_{k,i}\mathbb{1}\left(\left\|h^{(L)}_k\right\| \le R\right)$ and $S^{(L)}_{k,i} := \sum_{k'=0}^{k}\widetilde{X}^{(L)}_{k',i}$ so that $\left\{S^{(L)}_{k,i} : k = -1, 0, \dots, L-1\right\}$ is a $(\mathcal{G}_k)$–martingale. Hence, by Doob's martingale inequality, we have

$$\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k<Lt} X^{(L)}_{k,i}\mathbb{1}_{\|h^{(L)}_k\|\le R}\right|^2\right] = \mathbb{E}\left[\sup_{0\le t\le\tau}\left|S^{(L)}_{\lfloor Lt\rfloor,i}\right|^2\right] \le 4\,\mathbb{E}\left[\left|S^{(L)}_{\lfloor L\tau\rfloor,i}\right|^2\right]. \tag{3.62}$$

Fix $k = 0, \dots, L-1$. For $i = 1, \dots, d$, we compute the following conditional expectation.

$$\begin{aligned}\mathbb{E}\left[\left(S^{(L)}_{k,i}\right)^2 \;\middle|\; \mathcal{G}_{k-1}\right] &= \mathbb{E}\left[\left(S^{(L)}_{k-1,i}\right)^2 + 2\widetilde{X}^{(L)}_{k,i}\sum_{k'=0}^{k-1}\widetilde{X}^{(L)}_{k',i} + \left(\widetilde{X}^{(L)}_{k,i}\right)^2 \;\middle|\; \mathcal{G}_{k-1}\right] \\ &= \left(S^{(L)}_{k-1,i}\right)^2 + \mathbb{E}\left[\left(\widetilde{X}^{(L)}_{k,i}\right)^2 \;\middle|\; \mathcal{G}_{k-1}\right]. \end{aligned} \tag{3.63}$$

The cross-term disappear as $\mathbb{E}\left[\widetilde{X}^{(L)}_{k,i} \;\middle|\; \mathcal{G}_{k-1}\right] = \mathbb{E}\left[X^{(L)}_{k,i} \;\middle|\; \mathcal{G}_{k-1}\right] \mathbb{1}\left(\|h^{(L)}_k\| \leq R\right) = 0$ by definition of $X^{(L)}_{k,i}$. Furthermore, conditionally on $\mathcal{G}_{k-1}$ and on $\{\|h^{(L)}_k\| \leq R\}$, observe that $X^{(L)}_{k,i}$ is the centered square of a normal random variable whose variance is $\mathcal{O}(L^{-1})$ uniformly in k by (3.18), so there exist $C_{R,1} > 0$ depending only on R such that

$$\sup_{0 \leq k < L} \mathbb{E}\left[\left(\widetilde{X}^{(L)}_{k,i}\right)^2 \;\middle|\; \mathcal{G}_{k-1}\right] \leq C_{R,1} L^{-2}.$$

Hence, plugging back into (3.62), we obtain

$$\mathbb{E}\left[\sup_{0 \leq t \leq \tau} \left| \sum_{k \leq Lt} X^i_k \mathbb{1}_{\|h^{(L)}_k\| \leq R} \right|^2\right] \leq 4 C_{R,1} L^{-1}. \tag{3.64}$$

For the second term involving $Y^{(L)}_{k,i}$, we explicitly compute the conditional expectation using the definition of V in (3.37) and the definition of Q in (3.22).

$$\begin{aligned} Y^{(L)}_{k,i} &= \mathbb{E}\left[\left(\Delta V^b_k\right)^2_i + \sum_{j,l=1}^{d} h^{(L)}_{k,j} h^{(L)}_{k,l} \left(\Delta V^A_k\right)_{ij} \left(\Delta V^A_k\right)_{il} \;\middle|\; \mathcal{G}_{k-1}\right] - Q_i\left(t_k, h^{(L)}_k\right)\Delta_L \\ &= \int_{t_k}^{t_{k+1}} \left(\mathbb{E}\left[\Sigma^b_{s,ii} \mid \mathcal{G}_{k-1}\right] + \sum_{j,l=1}^{d} h^{(L)}_{k,j} h^{(L)}_{k,l} \mathbb{E}\left[\Sigma^A_{s,ijil} \mid \mathcal{G}_{k-1}\right]\right) \mathrm{d}s - Q_i\left(t_k, h^{(L)}_k\right)\Delta_L \\ &= \int_{t_k}^{t_{k+1}} \left(\mathbb{E}\left[\Sigma^b_{s,ii} - \Sigma^b_{t_k,ii} \mid \mathcal{G}_{k-1}\right] + \sum_{j,l=1}^{d} h^{(L)}_{k,j} h^{(L)}_{k,l} \mathbb{E}\left[\Sigma^A_{s,ijil} - \Sigma^A_{t_k,ijil} \mid \mathcal{G}_{k-1}\right]\right) \mathrm{d}s. \end{aligned}$$

By Cauchy-Schwarz, Tonelli and (3.19) in Assumption 3.5 (*ii*) we obtain:

$$
\begin{aligned}
&\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k\le L(t\wedge\theta_R)} Y_{k,i}^{(L)}\mathbb{1}_{\left\|h_k^{(L)}\right\|\le R}\right|^2\right] \le \mathbb{E}\left[\left(\sum_{k=0}^{L-1}\left|Y_{k,i}^{(L)}\right|\mathbb{1}_{\left\|h_k^{(L)}\right\|\le R}\right)^2\right]\\
&\le \mathbb{E}\left[\left(\sum_{k=0}^{L-1}\int_{t_k}^{t_{k+1}}\left(\mathbb{E}\left[\left|\Sigma^b_{s,ii}-\Sigma^b_{t_k,ii}\right| \mid \mathcal{G}_{k-1}\right]+R^2\sum_{j,l=1}^{d}\mathbb{E}\left[\left|\Sigma^A_{s,ijil}-\Sigma^A_{t_k,ijil}\right| \mid \mathcal{G}_{k-1}\right]\right)\mathrm{d}s\right)^2\right]\\
&\le \left(\sum_{k=0}^{L-1}\int_{t_k}^{t_{k+1}}(1+R^2)M^{1/2}\left|s-t_{k_s}\right|^{\kappa/2}\mathrm{d}s\right)^2\\
&= M(1+R^2)^2\left(L\int_0^{1/L} r^{\kappa/2}\mathrm{d}r\right)^2 = \frac{M(1+R^2)^2}{(1+\kappa/2)^2}L^{-\kappa} =: C_{R,2}L^{-\kappa},
\end{aligned}
\tag{3.65}
$$

where $C_{R,2}>0$ depends only on R. Moving to the third term of (3.61) involving $J_{k,i}^{(L)}$, observe that

$$
\sup_{\|h\|\le R}\mathbb{E}\left[\left|\sum_{j=1}^{d}\int_{t_k}^{t_{k+1}}(\mathrm{d}V_t^A)_{ij}h_j\right|^2\right] \le R^2 d\sum_{j=1}^{d}\mathbb{E}\left[\left|\sum_{l,m=1}^{d}\int_{t_k}^{t_{k+1}}\left(q_s^A\right)_{ijlm}\left(\mathrm{d}B_s^A\right)_{lm}\right|^2\right] \le C_7R^2\Delta_L,
$$

$$
\mathbb{E}\left[\left|\int_{t_k}^{t_{k+1}}(\mathrm{d}V_t^b)_i\right|^2\right] = \mathbb{E}\left[\left|\sum_{l=1}^{d}\int_{t_k}^{t_{k+1}}\left(q_r^b\right)_{il}\left(\mathrm{d}B_s^b\right)_l\right|^2\right] \le C_8\Delta_L,
$$

for some $C_7, C_8>0$ independent of R since $\sup_{0\le t\le 1}\left\|\Sigma_t^A\right\|\le C_1$ and $\sup_{0\le t\le 1}\left\|\Sigma_t^b\right\|\le C_1$ almost surely. Then there exists $C_{R,3}>0$ depending only on R such that

$$
\sup_{\|h\|\le R}\mathbb{E}\left[\left|J_{k,i}^{(L)}\right|^2\mathbb{1}_{\left\|h_k^{(L)}\right\|\le R}\right] \le C_{R,3}L^{-3}.
\tag{3.66}
$$

Finally, we bound the fourth term of (3.61) using Cauchy-Schwarz, Assumption 3.3 and property (3.18) of the Itô processes:

$$
\sigma'''(\xi_i)^2\sup_{\|h\|\le R}\mathbb{E}\left[M_{k,i}^{(L)}(h)^6\right] \le m^2\, C_{R,4}L^{-3},
\tag{3.67}
$$

for some constant $C_{R,4}>0$ depending only on R. Combining the results in (3.64), (3.65), (3.66) and (3.67), there exists constants $C_{R,5}, C_{R,6}>0$ depending only on R such that

$$
\mathbb{E}\left[\sup_{0\le t\le\tau}\left|\sum_{k\le L(t\wedge\theta_R)} D_k^{(L),i}\left(h_k^{(L)}\right)\right|^2\right] \le \frac{C_{R,5}}{4d}L^{-\kappa}+\frac{C_{R,6}}{4d}L^{-1}.
\tag{3.68}
$$

Upper bound on ①. Given $s \in [0, T \wedge \theta_R)$, we have

$$\begin{aligned}\overline{H}^{(L)}_s - \widetilde{H}^{(L)}_s &= h^{(L)}_{k_s} - \left(h^{(L)}_{k_s} + \int_{t_{k_s}}^{s} \mu(s, \overline{H}^{(L)}_s)\mathrm{d}s + \int_{t_{k_s}}^{s} \left(\mathrm{d}V^A_s \overline{H}^{(L)}_s + \mathrm{d}V^b_s\right) \right) \\ &= -\mu\left(t_{k_s}, h^{(L)}_{k_s}\right)(s - t_{k_s}) - \left(V^A_s - V^A_{t_{k_s}}\right) h^{(L)}_{k_s} - \left(V^b_s - V^b_{t_{k_s}}\right) \qquad (3.69)\end{aligned}$$

by continuity of μ. Hence

$$\left\|\overline{H}^{(L)}_s - \widetilde{H}^{(L)}_s\right\|^2 \leq 3\left\|\mu\left(t_{k_s}, h^{(L)}_{k_s}\right)\right\|^2 (\Delta_L)^2 + 3\left\|h^{(L)}_{k_s}\right\|^2 \left\|V^A_s - V^A_{t_{k_s}}\right\|^2 + 3\left\|V^b_s - V^b_{t_{k_s}}\right\|^2 \qquad (3.70)$$

Now, from the local Lipschitz condition (3.44), for $\|h\| \leq R$ we have almost surely

$$\|\mu(s,h)\|^2 \leq 2\left(\|\mu(s,h) - \mu(s,0)\|^2 + \|\mu(s,0)\|^2\right) \leq 2\left(C_R \|h\|^2 + \|\mu(s,0)\|^2\right).$$

Combining the two previous inequalities we obtain

$$\left\|\overline{H}^{(L)}_s - \widetilde{H}^{(L)}_s\right\|^2 \leq 4\left(C_R \left\|h^{(L)}_{k_s}\right\|^2 + \|\mu(s,0)\|^2 + 1\right)\left(\Delta_L^2 + \left\|V^A_s - V^A_{t_{k_s}}\right\|^2 + \left\|V^b_s - V^b_{t_{k_s}}\right\|^2\right).$$

Hence, using (3.45) and the Lyapunov inequality [127], we get

$$\begin{aligned}&\mathbb{E}\int_0^{\tau\wedge\theta_R} \left\|\overline{H}^{(L)}_s - \widetilde{H}^{(L)}_s\right\|^2 \mathrm{d}s \\ \leq\ & \mathbb{E}\int_0^{\tau\wedge\theta_R} 4\left(C_R\left\|h^{(L)}_{k_s}\right\|^2 + \|\mu(s,0)\|^2 + 1\right)\left(\Delta_L^2 + \left\|V^A_s - V^A_{t_{k_s}}\right\|^2 + \left\|V^b_s - V^b_{t_{k_s}}\right\|^2\right)\mathrm{d}s \\ \leq\ & \int_0^{\tau} 4\,\mathbb{E}\left[\left(C_R\left\|h^{(L)}_{k_s}\right\|^2 + \|\mu(s,0)\|^2 + 1\right)\left(\Delta_L^2 + \left\|V^A_s - V^A_{t_{k_s}}\right\|^2 + \left\|V^b_s - V^b_{t_{k_s}}\right\|^2\right)\right]\mathrm{d}s \\ \leq\ & \int_0^1 4\left(C_R\,\mathbb{E}\left[\left\|h^{(L)}_{k_s}\right\|^2\right] + \|\mu(s,0)\|^2 + 1\right)\left(\Delta_L^2 + 2C_1\Delta_L + 2C_1\Delta_L\right)\mathrm{d}s \\ \leq\ & 4\left(C_R C_0^{2/p} + 1 + \int_0^1 \|\mu(s,0)\|^2\mathrm{d}s\right)\left(\Delta_L^2 + 4C_1\Delta_L\right). \qquad (3.71)\end{aligned}$$

Combining the results in (3.59), (3.68) and (3.71), we have in (3.57) that

$$\begin{aligned}\mathbb{E}\left[\sup_{0\leq t\leq\tau}\left\|\widetilde{H}^{(L)}_{\tau\wedge\theta_R} - H_{t\wedge\theta_R}\right\|^2\right] &\leq C'_R\left(C_R C_0^{2/p} + 1 + \int_0^1 \|\mu(s,0)\|^2\mathrm{d}s\right)\left(L^{-2} + 4C_1L^{-1}\right) \\ &+ \frac{(\widetilde{C}_0 + 1)C_M}{1+\kappa}L^{-\kappa} + \left(C_{R,5}L^{-\kappa} + C_{R,6}L^{-1}\right) + C'_R\int_0^\tau \mathbb{E}\left[\sup_{0\leq r\leq s}\left\|\widetilde{H}^{(L)}_{r\wedge\theta_R} - H_{r\wedge\theta_R}\right\|^2\right]\mathrm{d}s.\end{aligned}$$

Applying the Grönwall inequality,

$$\mathbb{E}\left[\sup_{0\leq t\leq\tau}\left\|\widetilde{H}^{(L)}_{\tau\wedge\theta_R} - H_{t\wedge\theta_R}\right\|^2\right] \leq C_9 C_{R,7} L^{-\min\{1,\kappa\}}\exp(C'_R), \qquad (3.72)$$

where C_9 is a universal constant independent of L, R and δ and $C_{R,7}$ is a constant only depending on R. Combining (3.72) with (3.51), we have

$$\mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\right] \le C_9 C_{R,7} L^{-\min\{1,\kappa\}}\exp(C'_R) + \frac{2^{p+1}\delta C}{p} + \frac{2(p-2)C}{p\delta^{2/(p-2)}R^p}. \tag{3.73}$$

Given any $\epsilon > 0$, we can choose $\delta > 0$ so that $\frac{2^{p+1}\delta C}{p} < \frac{\epsilon}{3}$, then choose R so that $\frac{2(p-2)C}{p\delta^{2/(p-2)}R^p} < \frac{\epsilon}{3}$, and finally choose L sufficiently large so that

$$C_9 C_{R,7} L^{-\min\{1,\kappa\}}\exp(C'_R) \le \frac{\epsilon}{3}.$$

Therefore in (3.73), we have,

$$\mathbb{E}\left[\sup_{0\le t\le 1}\|e_1(t)\|^2\right] \le \epsilon. \tag{3.74}$$

It remains to provide a uniform bound for $\overline{H} - \widetilde{H}$. Recall the relationship between $\widetilde{H}$ and $\overline{H}$ defined in (3.69): by (3.18) we have almost surely that

$$\begin{aligned}\left\|\overline{H}_s^{(L)} - \widetilde{H}_s^{(L)}\right\|^2 &\le 3\left(\left\|\mu\left(t_{k_s}, h_{k_s}^{(L)}\right)\right\|^2 (\Delta_L)^2 + \left\|h_{k_s}^{(L)}\right\|^2 \left\|V_s^A - V_{t_{k_s}}^A\right\|^2 + \left\|V_s^b - V_{t_{k_s}}^b\right\|^2\right) \\ &\le C_{10}\left(\left\|h_{k_s}^{(L)}\right\|^4 + \left\|h_{k_s}^{(L)}\right\|^2 + 1\right)(\Delta_L)^2 \\ &\quad + 3\left(\left\|h_{k_s}^{(L)}\right\|^2 \left\|V_s^A - V_{t_{k_s}}^A\right\|^2 + \left\|V_s^b - V_{t_{k_s}}^b\right\|^2\right).\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbb{E}\left[\sup_{0\le s\le 1}\left\|\overline{H}_s^{(L)} - \widetilde{H}_s^{(L)}\right\|^2\right] &\le C_{10}\left(\mathbb{E}\left[\sup_{0\le s\le 1}\left\|h_{k_s}^{(L)}\right\|^4\right] + \mathbb{E}\left[\sup_{0\le s\le 1}\left\|h_{k_s}^{(L)}\right\|^2\right] + 1\right)(\Delta_L)^2 \\ + 3\Bigg(\Bigg(\mathbb{E}\left[\sup_{0\le s\le 1}\left\|h_{k_s}^{(L)}\right\|^4\right] &\mathbb{E}\left[\sup_{0\le s\le 1}\left\|V_s^A - V_{t_{k_s}}^A\right\|^4\right]\Bigg)^{1/2} + \mathbb{E}\left[\sup_{0\le s\le 1}\left\|V_s^b - V_{t_{k_s}}^b\right\|^2\right]\Bigg).\end{aligned} \tag{3.75}$$

First, by Assumption 3.7,

$$\mathbb{E}\left[\sup_{0\le s\le 1}\left\|h_{k_s}^{(L)}\right\|^n\right] = \mathbb{E}\left[\sup_{0\le s\le 1}\|\overline{H}_s^{(L)}\|^n\right] < \infty, \quad n \in \{2,4\}. \tag{3.76}$$

Second, by the Power Mean inequality and Doob's martingale inequality,

$$\begin{aligned}\mathbb{E}\left[\sup_{t_k\le s<t_{k+1}}\|V_s^A - V_{t_k}^A\|^4\right] &= \mathbb{E}\left[\sup_{t_k\le s<t_{k+1}}\left(\sum_{i,j=1}^d\left|\sum_{k,l=1}^d\int_{t_k}^s \left(q_r^A\right)_{ijkl}\left(\mathrm{d}B_r^A\right)_{kl}\right|^2\right)^2\right] \\ &\le d^8\sum_{i,j,k,l=1}^d \mathbb{E}\left[\sup_{t_k\le s<t_{k+1}}\left|\int_{t_k}^s \left(q_r^A\right)_{ijkl}\left(\mathrm{d}B_r^A\right)_{kl}\right|^4\right] \\ &\le \left(\frac{4}{3}\right)^4 d^8\sum_{i,j,k,l=1}^d \mathbb{E}\left[\left|\int_{t_k}^{t_{k+1}} \left(q_s^A\right)_{ijkl}\left(\mathrm{d}B_s^A\right)_{kl}\right|^4\right] \le C_{11}\Delta_L^2,\end{aligned}$$

Hence

$$\mathbb{E}\left[\sup_{0\le s\le 1}\left\|V_s^A - V_{t_{k_s}}^A\right\|^4\right] \le \mathbb{E}\left[\sum_{k=0}^{L-1}\left(\sup_{t_k\le s<t_{k+1}}\left\|V_s^A - V_{t_{k_s}}^A\right\|^4\right)\right] \le C_{11}\Delta_L. \tag{3.77}$$

By Hölder inequality,

$$\mathbb{E}\left[\sup_{0\le s\le 1}\left\|V_s^A - V_{t_{k_s}}^A\right\|^2\right] \le \left(\mathbb{E}\left[\sup_{0\le s\le 1}\left\|V_s^A - V_{t_{k_s}}^A\right\|^4\right]\right)^{1/2} \le \sqrt{C_{11}}\Delta_L^{1/2}. \tag{3.78}$$

Combining (3.76), (3.77), and (3.78) in (3.75), we obtain

$$\mathbb{E}\left[\sup_{0\le t\le 1}\|e_2(t)\|^2\right] = \mathbb{E}\left[\sup_{0\le t\le 1}\left\|\overline{H}_t^{(L)} - \widetilde{H}_t^{(L)}\right\|^2\right] \le C_{12}\Delta_L^{1/2},$$

for some constant $C_{12} > 0$. By choosing $L > (C_{12}/\epsilon)^2$, we have

$$\mathbb{E}\left[\sup_{0\le t\le 1}\|e_2(t)\|^2\right] \le \epsilon. \tag{3.79}$$

Finally, combining (3.74) and (3.79) leads to the desired result.

□

3.5 Asymptotic analysis of the backpropagation dynamics

The most widely used method to train neural networks is the pairing of

- the backpropagation algorithm to find the exact gradient (or a stochastic approximation) of the loss function with respect to the network weights, and
- a variant of the gradient descent algorithm to iteratively update the network weights.

We are interested to study the behaviour of the former in residual networks, under our Scaling regimes 1 and 2. To do so, we will first formalize the objective function and the discrete backward equation linking the gradient of the loss function across layers.

3.5.1 Backpropagation in supervised learning

Suppose we want to learn the mapping $f_{\text{true}} \in \mathcal{C}^1(\mathbb{R}^d, \mathbb{R}^d)$ through a dataset of input-target pairs $\mathcal{D} := \{(x_i, y_i) : i = 1, \dots, N\} \subset \mathbb{R}^d \times \mathbb{R}^d$, where $x_i \in B$ for some $B \subset \mathbb{R}^d$ compact and $y_i = f_{\text{true}}(x_i)$. The goal of any parametric supervised learning is to find, given a class of mappings $\phi_\theta : \mathbb{R}^d \to \mathbb{R}^d$, the parameter $\theta \in \Theta$ that minimizes the average training error:

$$J_{\mathfrak{D}}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell(\phi_\theta(x_i), y_i) = \frac{1}{N}\sum_{i=1}^{N} \ell(\phi_\theta(x_i), f_{\text{true}}(x_i)). \tag{3.80}$$

Here, $\ell : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ is a loss function, for example the squared error $\ell(\widehat{y}, y) = \|y - \widehat{y}\|^2$. In the following, we omit the dependence in $\mathfrak{D}$. Fix $L \in \mathbb{N}$ and define

$$\theta^{(L)} := \left(A_k^{(L)}, b_k^{(L)}\right)_{k=1}^{L} \in \left(\mathbb{R}^{d\times d} \times \mathbb{R}^d\right)^L.$$

For an input $x \in \mathbb{R}^d$, recall the following forward dynamics for the residual network

$$\begin{aligned} h_0^{(L),x} &= x, \\ h_{k+1}^{(L),x} &= h_k^{(L),x} + L^{-\alpha}\, \sigma_d\left(A_k^{(L)} h_k^{(L),x} + b_k^{(L)}\right). \end{aligned} \tag{3.81}$$

We define $\phi_{\theta^{(L)}}(x) := h_L^{(L),x}$. Our goal is to compute $\nabla_{\theta^{(L)}} J\left(\theta^{(L)}\right)$. Observe from the definition (3.80) and the chain rule that

$$\nabla_{\theta_k^{(L)}} J\left(\theta^{(L)}\right) = \frac{1}{N}\sum_{i=1}^{N} \nabla_{\theta_k^{(L)}} h_{k+1}^{(L),\,x_i} \left(\frac{\partial h_L^{(L),\,x_i}}{\partial h_{k+1}^{(L),\,x_i}} \frac{\partial \ell}{\partial \widehat{y}}\left(h_L^{(L),\,x_i}, y_i\right)\right).$$

The terms $\partial\ell/\partial\widehat{y}$ and $\nabla_{\theta_k} h_{k+1}$ are straightforward to obtain, so the crux of the challenge lies in computing $\partial h_L/\partial h_{k+1}$. Using (3.81), for $x \in \mathbb{R}^d$, we get

$$\begin{aligned} g_k^{(L),x} := \frac{\partial h_L^{(L),\,x}}{\partial h_k^{(L),x}} &= \frac{\partial h_L^{(L),\,x}}{\partial h_{k+1}^{(L),x}} \frac{\partial h_{k+1}^{(L),x}}{\partial h_k^{(L),x}} \\ &= g_{k+1}^{(L),x}\left(I_d + L^{-\alpha}\,\mathrm{diag}\left(\sigma_d'\left(A_k^{(L)} h_k^{(L),x} + b_k^{(L)}\right)\right) A_k^{(L)}\right), \end{aligned} \tag{3.82}$$

where $\sigma_d'(z) = (\sigma'(z_i))_{i=1}^d \in \mathbb{R}^d$ for $z \in \mathbb{R}^d$. The terminal condition is given by $g_L^{(L),x} = I_d$. We now obtain the asymptotic dynamics of g under three different cases. In particular, we derive (backward) ODE limits for any set of weights under Scaling regime 1 and the asymptotic limit derived from an SDE under Scaling regime 2. For clarity, we omit the dependence in the input x for $g_k^{(L)}$.

3.5.2 Backward equation for the Jacobian under Scaling regime 1

Let $\overline{G}^{(L)} : [0,1] \to \mathbb{R}^{d\times d}$ be a continuous-time extension of the Jacobians $g_k^{(L)}$ defined in (3.82):

$$\overline{G}_t^{(L)} = g_{k+1}^{(L)} \mathbb{1}_{\frac{k}{L} < t \leq \frac{k+1}{L}}, \quad k = 0, 1, \ldots, L-1. \tag{3.83}$$

Theorem 3.13 (Backpropagation limits under Scaling regime 1)**.** *Under the same assumptions as Theorem 3.4,*

- *Neural ODE regime: If $\alpha = 1$, $\beta = 0$, and $(H_t)_{t\in[0,1]}$ is the solution to the neural ODE (3.12), then the backpropagation dynamics converge uniformly to the solution to the linear (backward) ODE*

$$\frac{\mathrm{d}G_t}{\mathrm{d}t} = -G_t \mathrm{diag}\left(\sigma'_d\left(\overline{A}_t H_t + \overline{b}_t\right)\right)\overline{A}_t, \quad G_1 = I_d \tag{3.84}$$

  in the sense that $\lim_{L\to\infty}\sup_{0\le t\le 1}\|G_t - \overline{G}_t^{(L)}\| = 0$.

- *Linear ODE regime: If $\alpha + \beta = 1$, $\beta > 0$, and $(H_t)_{t\in[0,1]}$ is the solution to the linear ODE (3.13), then the backpropagation dynamics converge uniformly to the solution to the linear (backward) ODE*

$$\frac{\mathrm{d}G_t}{\mathrm{d}t} = -G_t\overline{A}_t, \quad G_1 = I_d \tag{3.85}$$

  in the sense that $\lim_{L\to\infty}\sup_{0\le t\le 1}\|G_t - \overline{G}_t^{(L)}\| = 0$.

The ideas of the proof follow closely those of Theorem 3.4 and the complete proof is given in Section 3.5.4.1. We readily see that under Scaling regime 1, the backward dynamics of the gradient become linear. When $\beta > 0$, which is the case observed in practice, the dependence on the activation function disappears in the large depth limit, exactly as for the forward dynamics.

3.5.3 Backward equation for the Jacobian under Scaling regime 2

Recall the set-up of Theorem 3.10. Let $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ be a probability space with a $\mathbb{P}$-complete filtration $\mathbb{F} = (\mathcal{F}_t)_{t\ge 0}$. Let $(B_t^A)_{t\ge0}$, resp. $(B_t^b)_{t\ge0}$, be $d\times d$-dimensional, resp. d-dimensional, independent $\mathbb{F}$-Brownian motions. Recall that for Scaling regime 2,

$$A_k^{(L)} = \overline{A}_{k/L}L^{-1} + W^A_{(k+1)/L} - W^A_{k/L} \qquad b_k^{(L)} = \overline{b}_{k/L}L^{-1} + W^b_{(k+1)/L} - W^b_{k/L}, \tag{3.86}$$

where $(W_t^A)_{t\in[0,1]}$ and $(W_t^b)_{t\in[0,1]}$ are Itô processes [131] adapted to $\mathbb{F}$ and can be written in the form:

$$\begin{aligned}\left(\mathrm{d}W_t^A\right)_{ij} &= \sum_{k,l=1}^{d}\left(q_t^A\right)_{ijkl}\left(\mathrm{d}B_t^A\right)_{kl} \quad \text{for } i,j = 1,\dots,d,\\ \mathrm{d}W_t^b &= q_t^b \mathrm{d}B_t^b,\end{aligned} \tag{3.87}$$

with $W_0^A = 0$, $W_0^b = 0$, $q_t^A \in \mathbb{R}^{d,\otimes 4}$ and $q_t^b \in \mathbb{R}^{d\times d}$ for $t \in [0,1]$. We use the notation in (3.16) and (3.17) for the quadratic variation of W^A and W^b.

Define

$$\nu(t,h) := \overline{A}_t \mathbb{1}_{\beta=1} + \frac{1}{2}\sigma''(0)\nabla_h Q(t,h). \tag{3.88}$$

We will use the following assumption for the results in this section:

Assumption 3.14.

$$\sup_L \mathbb{E}\left[\sup_{0\le t\le 1} \left\|\overline{G}_t^{(L)}\right\|^4\right] < \infty, \qquad \mathbb{E}\left[\exp\left(8\int_0^1 \left|\operatorname{tr}\left(\nu(s,H_s)\right)\right| \mathrm{d}s\right)\right] < \infty.$$

The boundedness of the fourth moment of the Jacobians $g_k^{(L)}$ in L is similar to Assumption (3.7) and is standard in the convergence of approximation schemes for SDE. The second part of Assumption 3.14 is a technical condition: we need the fourth moment of the $\nabla_x H_t^x$ to be bounded. Theorem 3.15 proves that the process $t \mapsto \nabla_x H_t^x$ satisfy a linear SDE with drift $\nu(t,H_t)$ linear in H_t, so we need finiteness of the L^8 norm of the exponential of the drift, see Lemma D.6 for more details. In practice, $g_k^{(L)}$ and $h_k^{(L),x}$ stay bounded during training, so Assumption (3.14) is satisfied.

Theorem 3.15 (Backpropagation dynamics under Scaling regime 2)**.** *Let Assumptions 3.3, 3.5, 3.7, and 3.14 hold and let $\alpha = 0$ and $\beta = 1$. Let $(H_t)_{t\in[0,1]}$ be a solution to the SDE (3.28) and $(J_t)_{t\in[0,1]} \subset \mathbb{R}^{d\times d}$ be the unique solution to the linear matrix-valued SDE*

$$\mathrm{d}J_t = \left(\nu(t,H_t)\mathrm{d}t + \mathrm{d}W_t^A\right)J_t, \quad J_0 = I_d, \tag{3.89}$$

where ν is defined in (3.88). Then, $\mathbb{P}-$a.s., J_t is invertible for all $t \in [0,1]$ and

$$\overline{G}^{(L)} = \sum_{k=0}^{L-1} g_k^{(L)} \mathbb{1}_{[t_k,t_{k+1})} \overset{L\to\infty}{\longrightarrow} G_t := J_1 J_t^{-1} \tag{3.90}$$

uniformly in $L^1(\mathbb{P})$ in the sense of Def. 3.8.

The steps to prove Theorem 3.15 are similar to those of Theorem 3.10 but the details are technically more involved. Indeed, terms that depend on $g_k^{(L),x}$ are not a priori adapted to the filtration generated by the Ito processes W^A and W^b. To overcome this challenge, we denote

$$J_k^{(L),x} := \nabla_x h_k^{(L),x}, \tag{3.91}$$

and we can rewrite $g_0^{(L),x} = g_k^{(L),x} J_k^{(L),x}$. This leads to a new perspective to understand $g_k^{(L),x}$ through two components $J_k^{(L),x}$ and $g_0^{(L),x}$. The first term $J_k^{(L),x}$ is adapted to the filtration generated by the Ito processes, and $g_0^{(L),x}$ is the Jacobian of the output with respect to the input, and does not depend on the layer. The complete proof is provided in Section 3.5.4.2.

3.5.3.1 Connection with Neural SDE

In a recent work, [98] show that, when the hidden state H satisfies a continuous-time 'neural SDE' dynamics, the Jacobian of the output with respect to the hidden states satisfies a backward SDE:

$$\mathrm{d}\widehat{G}_t = \widehat{G}_t \left(-\nu(t, \widehat{H}_t)\mathrm{d}t - \mathrm{d}\widehat{W}_t^A\right), \quad \widehat{G}_1 = I_d, \tag{3.92}$$

where $\widehat{W}^A$ is the time-reversed Brownian motion defined by $\widehat{W}_t^A := W_t^A - W_1^A$, and $\widehat{H}_t$ is the solution of the backward flow of diffeomorphisms generated by the forward SDE (3.28).

It is clear that the limit G_t in (3.90) differs from the adjoint process (3.92). Our limit $G_t = J_1 J_t^{-1}$ does not satisfy any forward nor backward SDE, as its solution is a function of H_1 which depends on weights across all layers i.e. the entire path of W^A. Indeed, Theorem 3.1 in [157] states that $t \mapsto J_t^{-1}$ solves the following SDE.

$$\mathrm{d}(J_t^{-1}) = J_t^{-1}\left(-\nu(t, H_t)\mathrm{d}t - \mathrm{d}W_t^A + \mathrm{d}\left[W^A\right]_t\right), \quad J_0^{-1} = I_d.$$

Therefore, one can write

$$\begin{aligned} G_t = J_1 J_t^{-1} &= J_1 \left(J_1^{-1} + \int_t^1 J_s^{-1}\left(-\nu(s, H_s)\mathrm{d}s - \mathrm{d}W_s^A + \mathrm{d}\left[W^A\right]_s\right)\right) \\ &= I_d + \int_t^1 G_s\left(-\nu(s, H_s)\mathrm{d}s - \mathrm{d}W_s^A + \mathrm{d}\left[W^A\right]_s\right). \end{aligned} \tag{3.93}$$

One can readily see that G_t depends on H_1 for all $t \in [0,1]$. Note that the quadratic variation drift correction term stems from using Ito integrals instead of Stratonovitch integrals.

In contrast to (3.92), (3.93) is the exact large-depth limit of gradients computed by backpropagation in finite depth residual networks, as stated in Theorem 3.15.

3.5.4 Proofs

3.5.4.1 Proof of Theorem 3.13

The ideas of the proof follow closely those of Theorem 3.4, and we will provide here the main arguments to the Neural ODE case. The other case is very similar.

Denote $t_k = k/L$, $k = 0, 1, \dots, L$ as the uniform partition of the interval $[0, 1]$. For $t \in (t_k, t_{k+1}]$, define

$$\widetilde{G}_t^{(L)} \coloneqq g_{k+1}^{(L)} \left(I_d + (t_{k+1} - t) \operatorname{diag}\left(\sigma_d' \left(\overline{A}_{t_k} H_{t_k} + \overline{b}_{t_k} \right) \right) \overline{A}_{t_k} \right),$$

where $\overline{A}$ and $\overline{b}$ are specified in Theorem 3.4. Hence, we can directly deduce that

$$\begin{aligned}\sup_{t\in[0,1]} \left\| \widetilde{G}_t^{(L)} - \overline{G}_t^{(L)} \right\| &\leq L^{-1} \sup_{0\leq k<L} \sup_{t\in(t_k,t_{k+1}]} \left\| g_{k+1}^{(L)} \operatorname{diag}\left(\sigma_d' \left(\overline{A}_{t_k} H_{t_k} + \overline{b}_{t_k} \right) \right) \overline{A}_{t_k} \right\| \\ &\leq L^{-1} \sup_{0\leq k<L} \left\| g_{k+1}^{(L)} \right\| \sup_{t\in[0,1]} \left\| \operatorname{diag}\left(\sigma_d' \left(\overline{A}_t H_t + \overline{b}_t \right) \right) \overline{A}_t \right\|\end{aligned}$$

By continuity of $\overline{A}$, $\overline{b}$ and H, the first supremum is finite and by a similar argument as in the proof of Theorem 3.4, the second supremum is also finite. Thus, there exists a constant $G_\infty > 0$ such that $\sup_{t\in[0,1]} \left\| \widetilde{G}_t^{(L)} - \overline{G}_t^{(L)} \right\| \leq G_\infty L^{-1}$. Now, we also have, for $t \in (t_k, t_{k+1}]$,

$$\begin{aligned}\widetilde{G}_t^{(L)} - G_t = \widetilde{G}_{t_{k+1}}^{(L)} - G_{t_{k+1}} + (t_{k+1} - t)\, g_{k+1}^{(L)} \operatorname{diag}\left(\sigma_d' \left(\overline{A}_{t_k} H_{t_k} + \overline{b}_{t_k} \right) \right) \overline{A}_{t_k} \\ - \int_t^{t_{k+1}} G_s \operatorname{diag}\left(\sigma_d' \left(\overline{A}_s H_s + \overline{b}_s \right) \right) \overline{A}_s \mathrm{d}s.\end{aligned}$$

Hence, for $e_k^{(L)} \coloneqq \sup_{t_k<t\leq t_{k+1}} \left\| \widetilde{G}_t^{(L)} - G_t \right\|$, we can estimate

$$\begin{aligned}e_k^{(L)} &\leq e_{k+1}^{(L)} + \int_t^{t_{k+1}} \left\| G_s J_s - g_{k+1}^{(L)} J_{t_k} \right\| \mathrm{d}s. \\ &\leq e_{k+1}^{(L)} + \int_t^{t_{k+1}} \left(\left\| G_s - g_{k+1}^{(L)} \right\| \|J_s\| + \left\| g_{k+1}^{(L)} \right\| \|J_s - J_{t_k}\| \right) \mathrm{d}s. \\ &\leq e_{k+1}^{(L)} + J_\infty L^{-1} \left(e_k^{(L)} + G_\infty L^{-1} \right) + \int_t^{t_{k+1}} \left\| g_{k+1}^{(L)} \right\| \|J_s - J_{t_k}\| \,\mathrm{d}s,\end{aligned}$$

where $J_s \coloneqq \operatorname{diag}\left(\sigma_d' \left(\overline{A}_s H_s + \overline{b}_s \right) \right) \overline{A}_s$ and $J_\infty \coloneqq \sup_{s\in[0,1]} \|J_s\| < \infty$. Now, recall that $\left\| g_{k+1}^{(L)} \right\|$ is uniformly bounded in k, L, and we have $\overline{A}, \overline{b} \in \mathcal{H}^1$ and $H \in \mathcal{C}^1$, so there exists a constant $J_\infty' < \infty$ such that $\|g_{k+1}^{(L)}\| \int_t^{t_{k+1}} \|J_s - J_{t_k}\| \,\mathrm{d}s < J_\infty' L^{-2}$. Thus,

$$\left(1 - J_\infty L^{-1} \right) e_k^{(L)} \leq e_{k+1}^{(L)} + \left(J_\infty G_\infty + J_\infty' \right) L^{-2}.$$

By Gronwall's lemma and the fact that $e_L^{(L)} = \mathcal{O}(L^{-1})$, we deduce that $\max_k e_k^{(L)} = \mathcal{O}(L^{-1})$ and conclude

$$\lim_{L\to\infty} \sup_{t\in[0,1]} \left\| \widetilde{G}_t^{(L)} - G_t \right\| \leq \lim_{L\to\infty} \left(\sup_{t\in[0,1]} \left\| \widetilde{G}_t^{(L)} - \overline{G}_t^{(L)} \right\| + \max_k e_k^{(L)} \right) = 0.$$

3.5.4.2 Proof of Theorem 3.15

The ideas of the proof follow closely those of Theorem 3.10, and we will provide here the main arguments for the case $\alpha = 0$ and $\beta = 1$. Other cases follow similarly. For the ease of notation exposition, we consider $U = 0$ and we use C to denote some generic constant (independent from L and other parameters, such as ε, δ, and R, to be defined later) that may vary from step to step.

The proof consists of 11 steps that can be summarized as follows. Step 1 decomposes the discrete gradient $g_k^{(L)}$ into two terms: the Jacobian of the output with respect to the input, and the Jacobian of the hidden state $h_k^{(L)}$ with respect to the input, which we denote by $J_k^{(L)}$. We then write a forward equation for $J_k^{(L)}$. Step 2 defines a continuous-time approximation $\widetilde{J}_k^{(L)}$ and a continuous-time interpolation $\overline{J}_k^{(L)}$. Step 3 establishes a uniform bound $\mathcal{O}(L^{-1})$ between $\widetilde{J}_k^{(L)}$ and $\overline{J}_k^{(L)}$. Step 4 defines high-probability events under which the hidden states $h_k^{(L)}$ and the continuous-time limit J_t are uniformly bounded. Step 5 decomposes the difference between $\widetilde{J}^{(L)}$ and J with a drift term and an error term $D^{(L)}$, which can be further decomposed into a variance term $N^{(L)}$ and a Taylor remainder term $R^{(L)}$. Step 6 proves that $R^{(L)}$ uniformly vanishes as $\mathcal{O}(L^{-1})$. Step 7 decomposes $N^{(L)}$ into three terms. Step 8 and 9 prove that these terms uniformly vanishes as $\mathcal{O}(L^{-\min(1,\kappa)})$. Step 10 wraps everything together to show a uniform L^2 bound between $\widetilde{J}^{(L)}$ and $J^{(L)}$. Step 11 uses it to prove a uniform L^1 bound between the discrete gradients $g_k^{(L)}$ and their limit $G_t = J_1 J_t^{-1}$.

Step 0: Well-posedness of the statement. The matrix-valued linear stochastic differential equation (3.89) has a continuous and adapted solution, and this solution is unique in the sense that almost all sample processes of any two solutions coincide, see for example [51]. Furthermore, $\mathbb{P}-$a.s., J_t is invertible for all $t \in [0, 1]$, see Corollary 2.1 in [42]. Also, Theorem 3.1 in [157] states that $K_t := J_t^{-1}$ solves the following SDE.

$$\mathrm{d}K_t = K_t \left(-\nu(t, H_t)\mathrm{d}t - \mathrm{d}W_t^A + \mathrm{d}\left[W^A\right]_t\right), \quad K_0 = I_d.$$

Recall from Assumption 3.5 that the quadratic variation of W^A is uniformly continuous with resepect to the Lesbegue measure. Therefore, by Lemma D.6 and Assumption 3.14, we conclude that the fourth moments of the supremum of J and J^{-1} are finite.

$$\mathbb{E}\left[\sup_{t\in[0,T]} \max\left(\left\|J_t^{-1}\right\|_F, \|J_t\|_F\right)^4\right] \le C\,\mathbb{E}\left[\exp\left(8\int_0^T |\mathrm{tr}\left(\nu(s, H_s)\right)| \,\mathrm{d}s\right)\right]^{1/2} < \infty. \tag{3.94}$$

Step 1: Rewrite the discrete backpropagation equation. First, observe that multiplying (3.82) together gives, for $k = 0, \dots, L$,

$$g_0^{(L),x} = g_k^{(L),x} \left[\prod_{k'=k-1}^{0} \left(I_d + \operatorname{diag}\left(\sigma_d' \left(A_{k'}^{(L)} h_{k'}^{(L),x} + b_{k'}^{(L)} \right) \right) A_{k'}^{(L)} \right) \right]. \tag{3.95}$$

Define $J_0^{(L),x} \coloneqq I_d$ and for $k = 0, \dots, L-1$:

$$J_{k+1}^{(L),x} \coloneqq \left(I_d + \operatorname{diag}\left(\sigma_d' \left(A_k^{(L)} h_k^{(L),x} + b_k^{(L)} \right) \right) A_k^{(L)} \right) J_k^{(L),x}. \tag{3.96}$$

Note that by the chain rule, we directly have $J_k^{(L),x} = \nabla_x h_k^{(L),x}$ and $g_0^{(L),x} = g_k^{(L),x} J_k^{(L),x}$. In the following, we omit the explicit dependence on the initial data x when the context is clear. Recall now the definition $M_k^{(L)}(h) \coloneqq A_k^{(L)} h + b_k^{(L)}$ from (3.41). By Taylor's theorem on σ', as σ''' is continuous, for each $i = 1, \dots, d$, there exists $\left| \xi_{k,i}^{(L)} \right| < \left| M_k^{(L)} \left(h_k^{(L)} \right)_i \right|$ such that

$$\sigma' \left(M_k^{(L)} \left(h_k^{(L)} \right)_i \right) = \sigma'(0) + \sigma''(0) M_k^{(L)} \left(h_k^{(L)} \right)_i + \frac{1}{2} \sigma''' \left(\xi_{k,i}^{(L)} \right) M_k^{(L)} \left(h_k^{(L)} \right)_i^2.$$

Hence, using $\sigma'(0) = 1$, $\Delta_L = L^{-1}$, and (3.86), we get

$$\begin{aligned}
J_{k+1}^{(L)} &= \left(I_d + \operatorname{diag}\left(\sigma_d' \left(M_k^{(L)} \left(h_k^{(L)} \right) \right) \right) A_k^{(L)} \right) J_k^{(L)} \\
&= \left(I_d + A_k^{(L)} + \sigma''(0) \operatorname{diag}\left(M_k^{(L)} \left(h_k^{(L)} \right) \right) A_k^{(L)} \right) J_k^{(L)} \\
&\quad + \operatorname{diag}\left(\frac{1}{2} \sigma''' \left(\xi_k^{(L)} \right) \odot M_k^{(L)} \left(h_k^{(L)} \right)^{\odot, 2} \right) A_k^{(L)} J_k^{(L)} \\
&= \left(I_d + \underbrace{\left(\overline{A}_{t_k} + \frac{1}{2} \sigma''(0) \nabla_h Q\left(t_k, h_k^{(L)} \right) \right)}_{=: \nu\left(t_k, h_k^{(L)} \right)} \Delta_L + \Delta W_k^A \right) J_k^{(L)} \\
&\quad + \left(\sigma''(0) \underbrace{\left(\operatorname{diag}\left(M_k^{(L)} \left(h_k^{(L)} \right) \right) A_k^{(L)} - \frac{1}{2} \nabla_h Q\left(t_k, h_k^{(L)} \right) \Delta_L \right)}_{=: N_k^{(L)}\left(J_k^{(L)}, h_k^{(L)} \right)} \right) J_k^{(L)} \\
&\quad \underbrace{+ \left(\operatorname{diag}\left(\frac{1}{2} \sigma''' \left(\xi_k^{(L)} \right) \odot M_k^{(L)} \left(h_k^{(L)} \right)^{\odot, 2} \right) \right) A_k^{(L)} J_k^{(L)}}_{=: R_k^{(L)}\left(J_k^{(L)}, h_k^{(L)} \right)} \\
&= J_k^{(L)} + \nu\left(t_k, h_k^{(L)} \right) J_k^{(L)} + \Delta W_k^A J_k^{(L)} + D_k^{(L)} \left(J_k^{(L)}, h_k^{(L)} \right),
\end{aligned} \tag{3.97}$$

where we define the error term $D_k^{(L)} \coloneqq \sigma''(0) N_k^{(L)} J_k^{(L)} + R_k^{(L)}$.

Step 2: Continuous-time approximation. Recall the (forward) SDE defined in the statement of the theorem

$$\mathrm{d}J_t = \nu(t, H_t)J_t\mathrm{d}t + \mathrm{d}W^A_t J_t, \quad J_0 = I_d.$$

Recall the definition of $\overline{H}^{(L)}$ in (3.40), and define similarly the *continuous-time extension* (CTE) of $\{J^{(L)}_k : k = 0, \dots, L\}$:

$$\overline{J}^{(L)}_t := \sum_{k=0}^{L} J^{(L)}_k \mathbf{1}_{t_k \leq t < t_{k+1}}. \tag{3.98}$$

Let k_s the index for which $t_{k_s} \leq s < t_{k_s+1}$. Define the *continuous-time approximation* (CTA) of J_t as

$$\widetilde{J}^{(L)}_t := I_d + \int_0^t \nu\left(t_{k_s}, \overline{H}^{(L)}_s\right) \overline{J}^{(L)}_s \mathrm{d}s + \int_0^t \mathrm{d}W^A_s \overline{J}^{(L)}_s + \sum_{k<Lt} D^{(L)}_k\left(J^{(L)}_k, h^{(L)}_k\right). \tag{3.99}$$

Step 3: Uniform bound between $\overline{J}^{(L)}$ and $\widetilde{J}^{(L)}$. Using (3.97) and (3.99), we have, for $s \in [0, 1]$,

$$\begin{aligned}\left\|\widetilde{J}^{(L)}_s - \overline{J}^{(L)}_s\right\|^2 &= \left\|\left(\nu\left(t_{k_s}, h^{(L)}_{k_s}\right)(s - t_{k_s}) + \left(W^A_s - W^A_{t_{k_s}}\right)\right) J^{(L)}_{k_s}\right\|^2 \\ &\leq \left(C\left(1 + \sup_k \left\|h^{(L)}_k\right\|^2\right)(\Delta_L)^2 + 2\left\|W^A_s - W^A_{t_{k_s}}\right\|^2\right)\left\|J^{(L)}_{k_s}\right\|^2\end{aligned}$$

Hence,

$$\begin{aligned}&\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\widetilde{J}^{(L)}_s - \overline{J}^{(L)}_s\right\|^2\right] \\ &\leq \left(C\left(1 + \mathbb{E}\left[\sup_k \left\|h^{(L)}_k\right\|^4\right]^{1/2}\right)(\Delta_L)^2 + \mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|W^A_s - W^A_{t_{k_s}}\right\|^4\right]^{1/2}\right)\mathbb{E}\left[\sup_k \left\|J^{(L)}_k\right\|^4\right]^{1/2}.\end{aligned}$$

By Assumptions 3.7 and 3.14, and equation (3.77):

$$\mathbb{E}\left[\sup_{0\leq s\leq 1}\left\|\widetilde{J}^{(L)}_s - \overline{J}^{(L)}_s\right\|^2\right] < CL^{-1}. \tag{3.100}$$

Step 4: Initial computations for a uniform L^1 bound between G and $\overline{G}^{(L)}$. Fix $\epsilon > 0$, and let $\delta > 0$ (to be determined later) that only depends on L and ϵ. Define for $R > 1$

$$E^{(L)}_R := \left\{\sup_{k\leq L}\left\|h^{(L)}_k\right\| \leq R\right\} \cap \left\{\sup_{t\in[0,1]}\|J_t\| \leq R\right\}. \tag{3.101}$$

Using Assumption 3.7 and (3.94), we obtain similarly to (3.54) that

$$\mathbb{P}\left((E_R^{(L)})^c\right) \leq \left(\mathbb{E}\left[\sup_{t\in[0,1]} \left\|\overline{H}_t\right\|^4\right] + \mathbb{E}\left[\sup_{t\in[0,1]} \left\|J_t\right\|^4\right]\right) R^{-4} \leq CR^{-4}. \tag{3.102}$$

Now, by Cauchy-Schwarz inequality, we have

$$ab = \delta^{1/2}a \cdot \delta^{-1/2}b \leq \frac{\delta}{2}a^2 + \frac{1}{2\delta}b^2, \quad \forall a, b, \delta > 0.$$

We use it to decompose the L^1 distance between G and $\overline{G}^{(L)}$:

$$\begin{aligned}\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\|\right] &= \mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\| \mathbb{1}_{E_R^{(L)}}\right] + \mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\| \mathbb{1}_{(E_R^{(L)})^c}\right] \\ &\leq \mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\| \mathbb{1}_{E_R^{(L)}}\right] + \frac{\delta}{2}\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\|^2\right] \\ &\quad + \frac{1}{2\delta}\mathbb{P}\left((E_R^{(L)})^c\right).\end{aligned}$$

Now, we have the following estimate

$$\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\|^2\right] \leq 2\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|J_t^{-1}\right\|^4\right]^{1/2} \mathbb{E}\left[\left\|J_1\right\|^4\right]^{1/2} + 2\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|\overline{G}_t^{(L)}\right\|^4\right]^{1/2}.$$

Therefore, by Assumption 3.14 and (3.94),

$$\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\|\right] \leq \mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|G_t - \overline{G}_t^{(L)}\right\| \mathbb{1}_{E_R^{(L)}}\right] + C\delta + \frac{C}{\delta R^4}. \tag{3.103}$$

Step 5: Initial computations for a uniform bound L^2 between J and $\widetilde{J}^{(L)}$. First we estimate, for $t \in [0,1]$,

$$\begin{aligned}\left\|J_t - \widetilde{J}_t^{(L)}\right\|^2 &\leq 3\int_0^t \left\|\nu\left(t_{k_s}, \overline{H}_s^{(L)}\right)\overline{J}_s^{(L)} - \nu\left(s, H_s\right)J_s\right\|^2 \mathrm{d}s + 3\left\|\int_0^t \mathrm{d}W_s^A\left(\overline{J}_s^{(L)} - J_s\right)\right\|^2 \\ &\quad + 3\left\|\sum_{k<Lt} D_k^{(L)}\left(J_k^{(L)}, h_k^{(L)}\right)\right\|^2.\end{aligned} \tag{3.104}$$

The goal is to find an upper bound of the first two terms, consisting of the sum of the L^2 distance between J and $\overline{J}^{(L)}$ and terms vanishing uniformly in L. We also want to show that the error term $D^{(L)}$ uniformly vanishes in L. To handle the term involving the drift ν, we first observe that for $t_1, t_2 \in [0,T]$, $h_1, h_2 \in \mathbb{R}^d$, and $J_1, J_2 \in \mathbb{R}^{d\times d}$, we have

$$\begin{aligned}\left\|\nu(t_2,h_2)J_2 - \nu(t_1,h_1)J_1\right\|^2 &\leq 3\left\|\nu(t_2,h_2)\right\|^2\left\|J_2 - J_1\right\|^2 + 3\left\|\nu(t_2,h_2) - \nu(t_2,h_1)\right\|^2\left\|J_1\right\|^2 \\ &\quad + 3\left\|\nu(t_2,h_1) - \nu(t_1,h_1)\right\|^2\left\|J_1\right\|^2. \\ &\leq C\left(1 + \left\|h_2\right\|^2\right)\left\|J_2 - J_1\right\|^2 + C\left\|h_2 - h_1\right\|^2\left\|J_1\right\|^2 \\ &\quad + C\left\|h_1\right\|^2\left\|J_1\right\|^2\left|t_2 - t_1\right|^{\kappa}.\end{aligned}$$

We used the fact that ν is linear in h and $\Sigma^A_{\cdot}$ is $\kappa/2-$Hölder continuous. We directly deduce that

$$\begin{aligned}
&\mathbb{E}\left[\mathbb{1}_{E_R^{(L)}} \int_0^T \left\|\nu\left(t_{k_s}, \overline{H}_s^{(L)}\right) \overline{J}_s^{(L)} - \nu\left(s, H_s\right) J_s\right\|^2 \mathrm{d}s\right] \\
&\leq C(1+R^2)\,\mathbb{E}\left[\mathbb{1}_{E_R^{(L)}} \int_0^T \left\|\overline{J}_s^{(L)} - J_s\right\|^2 \mathrm{d}s\right] + CR^2\,\mathbb{E}\left[\int_0^T \left\|\overline{H}_s^{(L)} - H_s\right\|^2 \mathrm{d}s\right] \qquad (3.105)\\
&\quad + CR^2\mathbb{E}\left[\int_0^T \left\|H_s\right\|^2 \mathrm{d}s\right] L^{-\kappa} \\
&\leq CR^2\,\mathbb{E}\left[\mathbb{1}_{E_R^{(L)}} \int_0^T \left\|\overline{J}_s^{(L)} - J_s\right\|^2 \mathrm{d}s\right] + CR^2 L^{-\min(1/2,\kappa)}. \qquad (3.106)
\end{aligned}$$

The last inequality holds by Theorem 3.10. Hence, we obtain from (3.104)

$$\begin{aligned}
\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|J_t - \widetilde{J}_t^{(L)}\right\|^2 \mathbb{1}_{E_R^{(L)}}\right] &\leq CR^2\,\mathbb{E}\left[\mathbb{1}_{E_R^{(L)}} \int_0^T \left\|\overline{J}_s^{(L)} - J_s\right\|^2 \mathrm{d}s\right] + CR^2 L^{-\min(1/2,\kappa)} \\
&\quad + 3\,\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|\sum_{k<Lt} D_k^{(L)}\left(J_k^{(L)}, h_k^{(L)}\right)\right\|^2 \mathbb{1}_{E_R^{(L)}}\right]. \qquad (3.107)
\end{aligned}$$

We applied Doob's martingale inequality [131] on the second term of (3.104), as $\overline{J}$, J, and $E_R^{(L)}$ are adapted to the filtration generated by W^A. We now estimate the error term $D^{(L)}$ in (3.107). Recall that it decomposes into a variance term $N^{(L)}$ and a Taylor remainder term $R^{(L)}$.

Step 6: Prove that the remainder $R_k^{(L)}$ uniformly vanishes. We proceed to show that

$$\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|\sum_{k<Lt} R_k^{(L)}\left(J_k^{(L)}, h_k^{(L)}\right)\right\|^2 \mathbb{1}_{E_R^{(L)}}\right] \leq CR^6 L^{-1}, \qquad (3.108)$$

which is straightforward since:

$$\begin{aligned}
&\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|\sum_{k<Lt} R_k^{(L)}\left(J_k^{(L)}, h_k^{(L)}\right)\right\|^2 \mathbb{1}_{E_R^{(L)}}\right] \leq \mathbb{E}\left[\left(\sum_{k=0}^{L-1}\left\|R_k^{(L)}\left(J_k^{(L)}, h_k^{(L)}\right)\right\|\right)^2 \mathbb{1}_{E_R^{(L)}}\right] \\
&\leq CR^2\,\mathbb{E}\left[\left(\sum_{k=0}^{L-1}\left(\left\|A_k^{(L)}\right\| R + \left\|b_k^{(L)}\right\|\right)^2 \left\|A_k^{(L)}\right\|\right)^2\right] \\
&\leq CR^2 L\,\mathbb{E}\left[\sum_{k=0}^{L-1}\left(\left\|A_k^{(L)}\right\| R + \left\|b_k^{(L)}\right\|\right)^4 \left\|A_k^{(L)}\right\|^2\right] \leq CR^6 L^{-1}.
\end{aligned}$$

The last inequality holds for the same reasons as (3.66).

Step 7: Prove that the remainder $N_k^{(L)}$ uniformly vanishes. First note that we can write $N_{k,ij}^{(L)} = N_{k,ij}^{(L),0} + \sum_{m=1}^{d} N_{k,ij,m}^{(L),1}$, where

$$\begin{aligned} N_{k,ij}^{(L),0} &:= b_{k,i}^{(L)} A_{k,ij}^{(L)} \\ N_{k,ij,m}^{(L),1} &:= \left(A_{k,im}^{(L)} A_{k,ij}^{(L)} - \left(\Sigma_{t_k}^{A}\right)_{imij} \Delta_L \right) h_{k,m}^{(L)} \end{aligned} \tag{3.109}$$

We assumed that the Ito processes W^A and W^b are driven by uncorrelated Brownian motions, hence $N_{k,ij}^{(L),0}$ uniformly vanishes in L^2 at rate Δ_L. Thus, we get

$$\begin{align} \mathbb{E}\left[\sup_{0\leq t\leq T} \left| \sum_{k<Lt} N_{k,ij}^{(L)} \right|^2 \mathbb{1}_{E_R^{(L)}} \right] \leq CL^{-1} + d \sum_{m=1}^{d} \mathbb{E}\left[\sup_{0\leq t\leq T} \left| \sum_{k<Lt} N_{k,ij,m}^{(L),1} \right|^2 \mathbb{1}_{E_R^{(L)}} \right]. \tag{3.110} \\ \tag{3.111} \end{align}$$

Using the discrete (forward) filtration $\{\mathcal{G}_k : k = -1, 0, \ldots, L-1\}$ defined in (3.60), we now expand (3.109) using the definition of Scaling regime 2.

$$\begin{aligned} N_{k,ij,m}^{(L),1} = & \underbrace{\left(\left(\Delta W_k^A\right)_{im} \left(\Delta W_k^A\right)_{ij} - \int_{t_k}^{t_{k+1}} \mathbb{E}\Big[\left(\Sigma_s^A\right)_{imij} \,\Big|\, \mathcal{G}_{k-1} \Big] \mathrm{d}s \right) h_{k,m}^{(L)}}_{\textcircled{1}} \\ & \underbrace{+ h_{k,m}^{(L)} \int_{t_k}^{t_{k+1}} \mathbb{E}\Big[\left(\Sigma_s^A - \Sigma_{t_k}^A\right)_{imij} \,\Big|\, \mathcal{G}_{k-1} \Big] \mathrm{d}s}_{\textcircled{2}} \\ & \underbrace{+ \Big[\left(\overline{A}_{t_k}\right)_{im} \left(\Delta W_k^A\right)_{ij} + \left(\overline{A}_{t_k}\right)_{ij} \left(\Delta W_k^A\right)_{im} \Big] h_{k,m}^{(L)} \Delta_L + \left(\overline{A}_{t_k}\right)_{im} \left(\overline{A}_{t_k}\right)_{ij} h_{k,m}^{(L)} (\Delta_L)^2}_{\textcircled{3}}. \end{aligned}$$

Step 8: Prove that the term $\textcircled{1}$ uniformly vanishes. Define

$$X_{k,ij,m}^{(L)} := \left(\Delta W_k^A\right)_{im} \left(\Delta W_k^A\right)_{ij} - \int_{t_k}^{t_{k+1}} \mathbb{E}\Big[\left(\Sigma_s^A\right)_{imij} \,\Big|\, \mathcal{G}_{k-1} \Big] \mathrm{d}s,$$

and $S_{k,ij,m}^{(L)} := \sum_{k'=0}^{k} X_{k',ij,m}^{(L)}$. Observe that $\{S_{k,ij,m}^{(L)} : k = 0, \ldots, L\}$ is a $(\mathcal{G}_k)-$martingale, where the filtration $\mathcal{G}_k$ is defined in (3.60). Hence, by Doob's martingale inequality, we have

$$\mathbb{E}\left[\sup_{0\leq t\leq T} \left| \sum_{k<Lt} X_{k,ij,m}^{(L)} \right|^2 \right] = \mathbb{E}\left[\sup_{0\leq t\leq T} \left| S_{\lfloor Lt \rfloor,ij,m}^{(L)} \right|^2 \right] \leq 4\,\mathbb{E}\left[\left| S_{\lfloor LT \rfloor,ij,m}^{(L)} \right|^2 \right]. \tag{3.112}$$

Fix $k = 0, \ldots, L-1$ and compute the following conditional expectation.

$$\begin{aligned} \mathbb{E}\left[\left(S_{k,ij,m}^{(L)}\right)^2 \,\middle|\, \mathcal{G}_{k-1} \right] &= \mathbb{E}\left[\left(S_{k-1,ij,m}^{(L)}\right)^2 + 2X_{k,ij,m}^{(L)} S_{k-1,ij,m}^{(L)} + \left(X_{k,ij,m}^{(L)}\right)^2 \,\middle|\, \mathcal{G}_{k-1} \right] \\ &= \left(S_{k-1,ij,m}^{(L)}\right)^2 + \mathbb{E}\left[\left(X_{k,ij,m}^{(L)}\right)^2 \,\middle|\, \mathcal{G}_{k-1} \right]. \end{aligned} \tag{3.113}$$

The cross-term disappear as $\mathbb{E}\Big[X^{(L)}_{k,ij,m}\ \Big|\ \mathcal{G}_{k-1}\Big] = 0$. Furthermore, conditionally on $\mathcal{G}_{k-1}$, observe that $\big(X^{(L)}_{k,ij,m}\big)^2$ is the variance of a product of two normal random variable with $O(L^{-1})$ variance, uniformly in k by (3.18), so

$$\sup_{0\leq k<L}\mathbb{E}\left[\left(X^{(L)}_{k,ij,m}\right)^2\ \middle|\ \mathcal{G}_{k-1}\right]\leq CL^{-2}.$$

Hence, plugging it back into (3.112), we obtain

$$\mathbb{E}\left[\sup_{0\leq t\leq T}\left|\sum_{k<Lt}\textcircled{1}\right|^2\mathbb{1}_{E_R^{(L)}}\right]\leq R^2\,\mathbb{E}\left[\sup_{0\leq t\leq T}\left|\sum_{k<Lt}X^{(L)}_{k,ij,m}\right|^2\right]\leq CR^2L^{-1}. \tag{3.114}$$

Step 9: Prove that the terms $\textcircled{2}-\textcircled{3}$ uniformly vanishes. The term $\textcircled{2}$ can be estimated directly using Cauchy-Schwarz, Tonelli, and (3.19):

$$\begin{aligned}\mathbb{E}\left[\sup_{0\leq t\leq T}\left|\sum_{k<Lt}\textcircled{2}\right|^2\mathbb{1}_{E_R^{(L)}}\right]&\leq R^2\,\mathbb{E}\left[\left(\sum_{k=0}^{L}\int_{t_k}^{t_{k+1}}\mathbb{E}\Big[\big|\big(\Sigma^A_s-\Sigma^A_{t_k}\big)_{imij}\big|\ \Big|\ \mathcal{G}_{k-1}\Big]\mathrm{d}s\right)^2\right]\\&\leq CR^2\left(\sum_{k=0}^{L}\int_{t_k}^{t_{k+1}}|s-t_{k_s}|^{\kappa/2}\,\mathrm{d}s\right)^2\leq CR^2L^{-\kappa}.\end{aligned} \tag{3.115}$$

The estimation for term $\textcircled{3}$ is straightforward and similar to (3.66):

$$\mathbb{E}\left[\sup_{0\leq t\leq T}\left|\sum_{k<Lt}\textcircled{3}\right|^2\mathbb{1}_{E_R^{(L)}}\right]\leq CR^2L^{-1}. \tag{3.116}$$

Step 10: Uniform bound between J and $\widetilde{J}^{(L)}$. From equations (3.108) (3.110), (3.114), (3.115), and (3.116), we deduce that

$$\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|\sum_{k<Lt}D^{(L)}_k\left(J^{(L)}_k,h^{(L)}_k\right)\right\|^2\mathbb{1}_{E_R^{(L)}}\right]\leq CR^6L^{-\min(1,\kappa)}. \tag{3.117}$$

We then plug (3.117) into (3.107), together with Tonelli's theorem, to get

$$\begin{aligned}\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|J_t-\widetilde{J}^{(L)}_t\right\|^2\mathbb{1}_{E_R^{(L)}}\right]&\leq CR^2\,\mathbb{E}\left[\mathbb{1}_{E_R^{(L)}}\int_0^T\left\|\overline{J}^{(L)}_s-J_s\right\|^2\mathrm{d}s\right]+CR^6L^{-\min(1/2,\kappa)}\\&\leq CR^2\,\mathbb{E}\left[\mathbb{1}_{E_R^{(L)}}\int_0^T\left\|\widetilde{J}^{(L)}_s-J_s\right\|^2\mathrm{d}s\right]+CR^6L^{-\min(1/2,\kappa)}.\end{aligned}$$

We use (3.100) for the last inequality. Hence, by Gronwall lemma, we deduce:

$$\mathbb{E}\left[\sup_{0\leq t\leq 1}\left\|J_t-\widetilde{J}^{(L)}_t\right\|^2\mathbb{1}_{E_R^{(L)}}\right]\leq CR^6L^{-\min(1/2,\kappa)}\exp\left(CR^2\right). \tag{3.118}$$

Step 11: Difference between G and g. We first estimate the L^1 distance between the discrete gradients $g_k^{(L)}$ and the continuous-time limit G_t. For each $t \in [0,1]$, we have the identity

$$\begin{aligned}\overline{G}_t^{(L)} - G_t &= \left(J_L^{(L)} - J_1\right) J_t^{-1} + g_{k_t}^{(L)} - J_L^{(L)} J_t^{-1} \\ &= \left(J_L^{(L)} - J_1\right) J_t^{-1} + g_{k_t}^{(L)} \left(J_t - J_{k_t}^{(L)}\right) J_t^{-1}.\end{aligned}$$

Hence, by Assumption 3.14, (3.94), (3.100), and (3.118):

$$\begin{aligned}&\mathbb{E}\left[\sup_{0\le t\le 1} \left\|G_t - \overline{G}_t^{(L)}\right\| \mathbb{1}_{E_R^{(L)}}\right] \\ &\le \mathbb{E}\left[\left\|\overline{J}_1^{(L)} - J_1\right\|^2 \mathbb{1}_{E_R^{(L)}}\right]^{1/2} \mathbb{E}\left[\sup_{0\le t\le 1} \left\|(J_t)^{-1}\right\|^2\right]^{1/2} \\ &\quad + \mathbb{E}\left[\sup_{0\le t\le 1} \left\|\overline{G}_t\right\|^4\right]^{1/4} \mathbb{E}\left[\sup_{0\le t\le 1} \left\|J_t - \overline{J}_t^{(L)}\right\|^2 \mathbb{1}_{E_R^{(L)}}\right]^{1/2} \mathbb{E}\left[\sup_{0\le t\le 1} \left\|(J_t)^{-1}\right\|^4\right]^{1/4}. \\ &\le C R^3 L^{-\min(1/4,\,\kappa/2)} \exp\left(C R^2\right).\end{aligned}$$

We plug it in (3.103) to obtain

$$\mathbb{E}\left[\sup_{0\le t\le 1} \left\|G_t - \overline{G}_t^{(L)}\right\|\right] \le C_1 R^3 L^{-\min(1/4,\,\kappa/2)} \exp\left(C_2 R^2\right) + C_3\delta + \frac{C_4}{\delta R^4}. \tag{3.119}$$

To conclude, given any $\epsilon > 0$, we can choose $\delta > 0$ such that $\delta < \frac{\epsilon}{3C_3}$, and then choose $R > 1$ so that $R^4 > \frac{3C_4}{\delta\epsilon}$, and finally L sufficiently large so that

$$C_1 R^3 L^{-\min(1/4,\,\kappa/2)} \exp\left(C_2 R^2\right) < \frac{\epsilon}{3}.$$

Therefore, we have in (3.119)

$$\mathbb{E}\left[\sup_{0\le t\le 1} \left\|G_t - \overline{G}_t^{(L)}\right\|\right] \le \epsilon.$$

Chapter 4

Convergence and implicit regularisation of gradient descent for deep residual networks

4.1 Introduction

Whether gradient descent methods find globally optimal solutions in the training of neural networks and how trained neural networks generalize are two major open questions in the theory of deep learning. The non-convexity of the loss functions for neural network training may lead to sub-optimal solutions when applying gradient descent methods. It is thus relevant to understand from a theoretical point of view whether specific neural network architectures with a proper choice of learning rates for gradient descent methods can improve the optimization landscape and/or eliminate sub-optimal solutions [146]. There is some empirical evidence that gradient descent seems to select solutions that generalize well [162] even without any *explicit* regularization. Hence, it is believed that gradient descent induces an implicit regularization [119] and characterizing the nature of this regularization is an interesting research question.
In the present work we prove linear convergence of gradient descent to a global minimum for a class of deep residual networks with constant layer width and smooth activation function. Furthermore, we show that under practical assumptions, the trained weights admit a scaling limit as a function of the layer index which has finite 2-variation. Our result shows that how implicit regularization emerges from gradient descent. Our proofs are based on non-asymptotic estimates for the loss function and norms of the network weights along the gradient descent path. These non-asymptotic estimates are interesting in their own right and may prove useful to other researchers for the study of dynamics of learning algorithms.

4.1.1 Convergence and regularization properties of deep learning algorithms

Existing results on convergence and implicit regularization in deep learning exploit three paradigms: over-parametrized neural networks with fixed depth and large width, linear neural networks with sufficiently large depth, and mean-field residual networks. Under sufficient over-parametrization by width with *fixed depth*, many popular neural network architectures (including feed-forward, convolutional, and residual) with ReLU activation find a global optimum in linear time with respect to the remaining error and the trained network generalizes well [3, 4]. However, the associated generalization bounds are intractable, and the amount of over-parametrization implied in these results is often unrealistically large. One can improve the asymptotic analysis [163, 164], but it still falls short of leading to any practical insight. For smooth activation functions, [44] studied the convergence of gradient descent for various network architectures, including residual networks. They show that for any depth, if the residual layers are wide enough and the learning rate is small enough, gradient descent on the empirical mean-squared loss converges to a solution with zero training loss in linear time. The rate of convergence is proportional to the learning rate and the minimum eigenvalue of the *Gram matrix*. [49] showed that in the over-parametrized regime, for a suitable initialization with the last layer initialized at zero and other weights initialized uniformly, gradient descent can find a global minimum exponentially fast with high probability.

For *linear* deep neural networks (i.e. with identity activation function), [15] showed that training with gradient descent is able to learn the positive definite linear transformations using identity initialization. [153] proposed a new initialization scheme named zero-asymmetric (ZAS) and proved that that under such initialization, for an arbitrary target matrix, gradient descent converges to an ϵ-optimal point in $\mathcal{O}(L^3 \log(1/\epsilon))$ iterations, which scales polynomially with the network depth L. Subsequent refinements of the convergence rates and the width requirements have been established in [43, 165]. Finally, [160] showed the implicit regularization of gradient descent for linear fully-connected networks to l_2 max-margin solutions.

Another line of work deals with mean-field residual networks by looking at the continuum limit of residual networks when either the depth L or the width d goes to infinity. [158] build on the analysis of [39] for feed-forward networks to study the average behaviour of randomly initialized residual networks with width tending to infinity. They show that a careful initialization, depending on the depth, may enhance

expressivity. Further, [106] proposed a continuum limit of deep residual networks by letting the depth L tends to infinity and showed that every local minimum of the loss landscape is global. This characterization enables them to derive the first global convergence result for multi-layer neural networks in the mean-field regime.
In addition to the network architectures listed above, non-linear neural networks with fixed width and *large but finite* depth are successful and practically more popular [71, 73]. It is well-documented that for a fixed number of parameters, going deeper allows the models to capture richer structures [50, 147]. However, the theoretical foundations for such networks remain widely open due to their complex training landscape.

4.1.2 Contributions

We consider a supervised learning problem where we seek to learn an unknown mapping with inputs and outputs in $\mathbb{R}^d$ using a residual network with constant width d and a smooth activation function. We study the convergence and implicit regularization of gradient descent for the mean-squared error.

- **Linear convergence.** For $\epsilon > 0$, we prove that for a residual network of depth $L = \Omega(1/\epsilon)$, we can choose a learning rate schedule such that gradient descent on the training loss converges to a ϵ-optimal solution in $\Theta(\log(1/\epsilon))$ iterations.

- **Scaling limit of trained weights.** The trained weights, as a function of the layer, may admit a scaling limit as $L \to \infty$. We prove that such a scaling limit is a matrix-valued function with finite 2-variation.

- **Non-asymptotic estimates on loss function and weights along the gradient descent path.** In addition to the convergence results mentioned above, we obtain (non-asymptotic) estimates along the gradient descent path for the loss function and various norms of the weights, with tractable bounds.

- **Relevance to practical settings.** We illustrate the relevance of our theoretical results in practical settings using detailed numerical experiments with networks of realistic width and depth.

Our analysis generalizes previous results on *linear* neural networks [153] to a more general nonlinear setting relevant for learning problems. Our non-asymptotic results stand in contrast to the mean-field analysis [106] which requires infinite depth. Our tractable bounds improve upon the ones found for networks over-parametrized by width [4, 44, 49, 163, 164], where the trained weights do not leave the lazy training regime [33]: in our setting, the trained weights are not necessarily staying close to their initialization. A key ingredient in the proof is to study the evolution of various norms for the weights under gradient descent iterations. These estimates are provided in Lemmas 4.4 and 4.5.

Our theoretical results suggest that initialization of weights at scale L^{-1} together with a $L^{-1/2}$ scaling of the activation function leads to convergence under a constant learning rate. The overarching principle is to make sure that the gradient stays on the same scale as the weights (here $L^{-1/2}$) during training. Our analysis also extends, with minimal changes, to the case where linear layers are added at the beginning and the end of the network.

Notations Define $(e_m)_{m'} = \mathbb{1}_{\{m'=m\}} \in \mathbb{R}^d$. For a vector $x \in \mathbb{R}^d$, we denote $\|x\|_2$ the Euclidean norm of x, and for a matrix $M \in \mathbb{R}^{d\times d}$, we denote $\|M\|_F$ the Frobenius norm of M. When the context is clear, we omit the superscript x for the quantities that depend on the input x. We denote $f = \mathcal{O}(g)$ if there exists $c > 0$ such that $f(z) \leq cg(z)$, where $z = (L, k, t, \eta_L(t), c_0)$. That means, our Big-O notation involves a constant that is independent of the depth L, the layer number k, the iteration number t, the learning rates $\eta_L(t)$, and the universal constant c_0 defined in Assumption 4.1. Similar definitions stand for Ω and Θ. For a function $\sigma\colon \mathbb{R} \to \mathbb{R}$, define $\sigma_d\colon \mathbb{R}^d \to \mathbb{R}^d$ by $\sigma_d(x)_i = \sigma(x_i)$ for $i = 1, \dots, d$.

4.2 Residual networks

Let $x \in \mathbb{R}^d$ be an input vector, δ_L be a fixed positive real number, and $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ be a set of parameters (or weights). In this section, we focus on a ResNet architecture *without bias* with L fully-connected layers:

$$\begin{cases} h_k^{x,(L)} &= h_{k-1}^{x,(L)} + \delta_L \sigma_d\left(\alpha_k^{(L)} h_{k-1}^{x,(L)}\right), \quad k = 1, \dots, L, \\ h_0^{x,(L)} &= x. \end{cases} \tag{4.1}$$

The output of the network is $h_L^{x,(L)}$, which we denote by $\widehat{y}_L\big(x, W^{(L)}\big)$ to emphasize the dependence on the input x and the weights $\alpha^{(L)}$. [1] Fix a training set $D_N := \{(x_i, y_i) : i = 1, \dots, N\} \subset \mathbb{R}^d \times \mathbb{R}^d$, and the loss function $\ell\colon \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ defined by $\ell(y, \widehat{y}) := \frac{1}{2}\|y - \widehat{y}\|_2^2$.

[1] The analysis with bias is done by expanding the weights $\alpha_k^{(L)}$ and the hidden states $h_k^{(L)}$ with an additional dimension.

We study the dynamics of the weights induced by gradient descent (GD) on the mean-squared error $J_L\colon \mathbb{R}^{L\times d\times d} \to \mathbb{R}_+$ defined by

$$J_L\big(\alpha^{(L)}\big) \coloneqq \frac{1}{N}\sum_{i=1}^{N} \ell\left(y_i, \widehat{y}_L\big(x_i, \alpha^{(L)}\big)\right) = \frac{1}{2N}\sum_{i=1}^{N} \left\|y_i - \widehat{y}_L\big(x_i, \alpha^{(L)}\big)\right\|_2^2 . \tag{4.2}$$

We consider a gradient descent learning algorithm which sequentially updates the weights using an initialization $A^{(L)}(0) \in \mathbb{R}^{L\times d\times d}$ and

$$\Delta A_k^{(L)}(t) \coloneqq A_k^{(L)}(t+1) - A_k^{(L)}(t) = -\eta_L(t)\nabla_{\alpha_k} J_L\big(A^{(L)}(t)\big)\,, \tag{4.3}$$

where $\eta_L(t) > 0$ is the *learning rate* at iteration $t \in \mathbb{N}$, which may depend on the depth L, but is independent of the layer index k.

Assumption 4.1. *There exists a constant $c_0 > 0$ such that*

(i) Smooth activation function: $\sigma \in C^2(\mathbb{R})$, $\sigma'(0) = 1$ and for all $z \in \mathbb{R}$, $|\sigma(z)| \le |z|$, $|\sigma'(z)| \le 1$ and $|\sigma''(z)| \le 1$.

(ii) Scaling factor: $\delta^{(L)} = L^{-1/2}$.

(iii) Separated unit data: $\|x_i\|_2 = \|y_i\|_2 = 1$ and $\forall i \neq j$, $|\langle x_i, x_j\rangle| \le (8N)^{-1}e^{-4c_0}$.

(iv) Initialisation with $O(1/L)$ weights:

$$\sup_{k,m}\left\|A_{k,m}^{(L)}(0)\right\|_2 \le 2^{-9/2}N^{-1/2}d^{-1/2}e^{-4.2c_0}L^{-1}.$$

(v) Small initial loss: $J_L(A^{(L)}(0)) \le 2^{-15}3^{-2}N^{-2}d^{-1}c_0^2e^{-8.2c_0}$.

Note that tanh satisfies Assumption 4.1 *(i)*. Assumption 4.1 *(ii)* comes from the scaling we observe in the experiments of Section 4.4.1. Assumption 4.1 *(iii)* requires the training points to be sufficiently orthogonal to one another. Among other cases, it is satisfied in the small data regime: take for example N points uniformly at random on the $d-$dimensional sphere, where $d > N^4$. Hence, for $x_i \sim \mathcal{U}(\mathbb{S}^{d-1})$ i.i.d., we have by a union bound and Chebychev inequality:

$$\begin{aligned}\mathbb{P}\left(\max_{i\neq j}|\langle x_i, x_j\rangle| > N^{-1}\right) &\le N^2\mathbb{P}\left(|\langle x_1, x_2\rangle| > N^{-1}\right)\\ &\le N^4\mathbb{V}\mathrm{ar}\left[\langle x_1, x_2\rangle\right] \le N^4\sum_{m=1}^{d}\mathbb{E}\left[(x_1)_m^2\right]\mathbb{E}\left[(x_2)_m^2\right] = N^4d^{-1} < 1.\end{aligned}$$

Assumption 4.1 *(iv)* guarantees that the network at initialization stay well-behaved, and does not bias the optimization path. Note also that Assumption 4.1 *(iv)* does not rule out the case of a stochastic initialization. Assumption 4.1 *(v)* relates to the fact that we are going to prove local convergence of gradient descent to zero training loss. Proving global convergence under our general framework is out of reach, as local minima are guaranteed to exist, see Theorem 2 in [126]. In this paper, we address Corollary 3 in [126] by providing conditions on the dataset and on the initialization procedure to show convergence of gradient descent for residual networks of large depth and finite width.

4.3 Dynamics of weights and hidden states under gradient descent

Recall that $\alpha^{(L)}$ denotes a generic weight vector, whereas $A^{(L)}(t)$ denotes the weight vector obtained after t iterations of gradient descent on the objective function J_L, where the initial weights $A^{(L)}(0)$ follow Assumption 4.1 *(iv)*. The main results can be summarized as follows.

First, in Section 4.3.1, we prove that if the network weights $\alpha_k^{(L)}$ are $\mathcal{O}(L^{-1/2})$, then the hidden states $h_k^{x,(L)}$ and the Jacobian

$$M_k^{x,(L)} := \frac{\partial h_L^{x,(L)}}{\partial h_k^{x,(L)}} \in \mathbb{R}^{d\times d} \tag{4.4}$$

are uniformly bounded in k and L. Then, under the same scaling assumption, we derive an upper bound for the norm of the gradient $\nabla_\alpha J_L$ of the objective function with respect to the weights $\alpha^{(L)}$. Furthermore, we derive a lower bound for the norm of the gradient $\nabla_\alpha J_L$ under the additional regularity assumption $\alpha_{k+1}^{(L)} - \alpha_k^{(L)} = \mathcal{O}(L^{-1})$.
Next, in Section 4.3.2, we let $\alpha^{(L)}(0) \in \mathbb{R}^{L\times d\times d}$ be *any* initialization and define recursively $\alpha^{(L)}(t+1) = \alpha^{(L)}(t) - \eta_L(t)\nabla_\alpha J_L\big(\alpha^{(L)}(t)\big)$. Under some scaling assumptions for $\alpha^{(L)}(t)$ for $t = 0,\dots,T-1$, we show that the loss function $J_L\big(\alpha^{(L)}(t)\big)$ at time T admits an explicit upper bound. To show this, we study the effect of gradient descent on the following norms of the weight vector:

$$\bar{f}^{(L)}\left(\alpha^{(L)}(t)\right) := \frac{1}{2}\sum_{k=1}^{L}\left\|\alpha_k^{(L)}(t)\right\|_F^2 \quad \text{and} \quad \bar{g}^{(L)}\left(\alpha^{(L)}(t)\right) := \frac{1}{2}L\sum_{k=1}^{L-1}\left\|\alpha_{k+1}^{(L)}(t) - \alpha_k^{(L)}(t)\right\|_F^2 . \tag{4.5}$$

The scaling in L is chosen in such a way that we will be able to prove a uniform bound (in t and L) of the above norms along the gradient descent path $A^{(L)}(t)$ when $A^{(L)}(0)$ satisfy Assumption 4.1 *(iii)*.

Finally in Section 4.3.3 we show that under Assumption 4.1 with the parameter $A^{(L)}(t)$ evolving according to the gradient descent dynamics (4.3), we have that for all $\epsilon > 0$, if we let $L = \Omega(1/\epsilon)$, $\eta_L(t) = \eta_0$, and $T_L^{\mathrm{const}} = \Theta(\eta_0^{-1} \log L) = \Omega(\eta_0^{-1} \log 1/\epsilon)$, then $J_L\big(A^{(L)}(T_L^{\mathrm{const}})\big) < \epsilon$. That is, the loss function can be made arbitrarily small with practical values for the depth and the number of gradient steps. To prove this, we use recursion: we first verify the scaling assumptions

$$A^{(L)}(t) = \mathcal{O}(c_0 L^{-1/2}) \quad \text{and} \quad A_{k+1}^{(L)}(t) - A_k^{(L)}(t) = \mathcal{O}(e^{-4.2c_0} L^{-1}), \tag{4.6}$$

at initialization, i.e. for $t = 0$. This enables us to use the results of Section 4.3.1 to deduce an upper bound on the loss function $J_L\big(A^{(L)}(1)\big)$ at time $t = 1$, which in turn yields that the scaling assumptions (4.6) are verified for $t = 1$. We continue this process until the upper bound on the loss is smaller than ϵ.
Further, we prove that for T_L satisfying (4.10), if the (pointwise) limit

$$\overline{A}_s^* := \lim_{L\to\infty} A_{\lfloor Ls \rfloor}^{(L)}(T_L) \tag{4.7}$$

converges uniformly in $s \in [0, 1]$ at a $\mathcal{O}(L^{-1/2})$ rate, then $\overline{A}^*$ is of finite 2-variation, giving an implicit regularity to the solution found by gradient descent. The numerical experiments in Section 4.4 confirm that these effects are observable in settings relevant to practical supervised learning problems.

4.3.1 Bounds on the hidden states, their Jacobians, and the loss gradients

We start the analysis by computing bounds on the hidden states and their Jacobians (4.4). To do so, we define the following norm on the weights:

$$\left\|\alpha^{(L)}\right\|_{F,\infty} := \max_{k=1,\dots,L} \left\|\alpha_k^{(L)}\right\|_F, \tag{4.8}$$

where $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ is a generic weight vector. We check that the hidden states are uniformly bounded from above and below in k and L, and we prove an upper bound on the Jacobians, uniformly in k and L. We get explicit bounds when L is large enough:

$$\|x\|_2 \, e^{-2c_A} \le \left\|h_k^{x,(L)}\right\|_2 \le \|x\|_2 \, e^{1.1c_A} \quad \text{and} \quad \left\|M_k^{x,(L)} e_m\right\|_2 \le e^{c_A},$$

given the assumption that $\left\|\alpha^{(L)}\right\|_{F,\infty} \le c_A L^{-1/2}$. The proof can be found in Appendix B.2. Note that the bounds are deterministic, unlike the probabilistic results from [4, 7]. Next, we derive that the norm of the gradient of the objective function is bounded above by $J_L^{1/2}$, so that it ensures that the gradient updates (4.3) stay local. The precise result and its proof can be found in Appendix B.3.

More crucially, we also need a lower bound on the norm of the gradient as a function of the *suboptimality* gap. We first establish a lower bound for the gradient of the loss with respect to the weights of the first layer.

Lemma 4.2. *Under Assumption 4.1 (i)–(iii), let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ such that $L \geq \max(5c_0, 4c_0^2)$ and $\left\|\alpha^{(L)}\right\|_{F,\infty} \leq c_0 L^{-1/2}$ hold. Then, we have*

$$\left\|\nabla_{\alpha_1} J_L\left(\alpha^{(L)}\right)\right\|_F^2 \geq \frac{1}{4N} e^{-2c_0} L^{-1} J_L\left(\alpha^{(L)}\right).$$

Proof. Fix $L \geq \max(5c_0, 4c_0^2)$. In the proof, we omit the explicit dependence in L. Observe first that

$$\begin{aligned}
\|\nabla_{\alpha_k} J_L(\alpha)\|_F^2 &= \sum_{m,n=1}^{d}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{\partial \ell}{\partial \alpha_{k,mn}}\left(y_i, \widehat{y}\left(x_i,\alpha\right)\right)\right)^2 \\
&= \sum_{m,n=1}^{d}\frac{\delta_L^2}{N^2}\sum_{i,j=1}^{N} h_{k-1,n}^{x_i} h_{k-1,n}^{x_j}\dot{\sigma}_{k,x_i,m}\dot{\sigma}_{k,x_j,m}\left(\left(M_k^{x_i}\right)^\top\left(\widehat{y}\left(x_i,\alpha\right)-y_i\right)\right)_m \\
&\qquad \left(\left(M_k^{x_j}\right)^\top\left(\widehat{y}\left(x_j,\alpha\right)-y_j\right)\right)_m \\
&= \frac{\delta_L^2}{N^2}\sum_{i,j=1}^{N}\left\langle h_{k-1}^{x_i}, h_{k-1}^{x_j}\right\rangle \widetilde{M}_{k,i,j},
\end{aligned}$$

where

$$\widetilde{M}_{k,i,j} = \left\langle \dot{\sigma}_{k,x_i}\odot\left(M_k^{x_i}\right)^\top\left(\widehat{y}\left(x_i,\alpha\right)-y_i\right),\ \dot{\sigma}_{k,x_j}\odot\left(M_k^{x_j}\right)^\top\left(\widehat{y}\left(x_j,\alpha\right)-y_j\right)\right\rangle.$$

We focus on the case $k=1$. We first estimate, by Cauchy-Schwarz and Lemma B.1,

$$\begin{aligned}
\left|\widetilde{M}_{1,i,j}\right| &\leq \left\|M_1^{x_i}\right\|_2\left\|M_1^{x_j}\right\|_2\left\|\widehat{y}\left(x_i,\alpha\right)-y_i\right\|_2\left\|\widehat{y}\left(x_j,\alpha\right)-y_j\right\|_2 \\
&\leq e^{2c_0}\left\|\widehat{y}\left(x_i,\alpha\right)-y_i\right\|_2\left\|\widehat{y}\left(x_j,\alpha\right)-y_j\right\|_2.
\end{aligned}$$

Lower bound when $i=j$ First, as $|\sigma''| \leq 1$ and $L \geq 4c_0^2$, we have $\dot{\sigma}_{1,x_i,m} = \sigma'(\alpha_1 x_i)_m \geq 1 - \|\alpha_1\|_F\|x_i\|_2 \geq 1 - c_0 L^{-1/2} \geq \frac{1}{2}$. Hence,

$$\begin{aligned}
\widetilde{M}_{1,i,i} &= \left\|\dot{\sigma}_{1,x_i}\odot\left(M_1^{x_i}\right)^\top\left(\widehat{y}\left(x_i,\alpha\right)-y_i\right)\right\|_2^2 \\
&\geq \frac{1}{4}\left\|\widehat{y}\left(x_i,\alpha\right)-y_i\right\|_2^2\prod_{k=1}^{L}\left(1-\delta_L\left\|\operatorname{diag}(\dot{\sigma}_{k,x_i})\alpha_k\right\|_2\right)^2 \\
&\geq \frac{1}{4}\left(1-\frac{c_0}{L}\right)^{2L}\left\|\widehat{y}\left(x_i,\alpha\right)-y_i\right\|_2^2 \geq \frac{1}{4}e^{-2c_0}\left\|\widehat{y}\left(x_i,\alpha\right)-y_i\right\|_2^2,
\end{aligned}$$

where we applied Lemma D.2 in the second line, and the fact that $\|\cdot\|_2 \leq \|\cdot\|_F$. By Assumption 4.1 *(iii)*, $|\langle x_i, x_j\rangle| \leq (8N)^{-1}e^{-4c_0}$ for all $i \neq j$, so we deduce

$$\begin{aligned}
\|\nabla_{\alpha_1} J_L(\alpha)\|_F^2 &= \frac{\delta_L^2}{N^2}\left(\sum_{i=1}^N \widetilde{M}_{1,i,i}\,\|x_i\|_2^2 + \sum_{i\neq j}\widetilde{M}_{1,i,j}\langle x_i, x_j\rangle\right) \\
&\geq \frac{1}{LN^2}\left(\frac{N}{2}e^{-2c_0}J_L(\alpha) - \frac{1}{8N}e^{-4c_0}\sum_{i\neq j}\left|\widetilde{M}_{1,i,j}\right|\right) \\
&\geq \frac{1}{LN^2}\left(\frac{N}{2}e^{-2c_0}J_L(\alpha) - \frac{1}{8N}e^{-2c_0}\sum_{i\neq j}\|\widehat{y}\,(x_i,\alpha) - y_i\|_2\,\|\widehat{y}\,(x_j,\alpha) - y_j\|_2\right) \\
&\geq \frac{1}{LN^2}\left(\frac{N}{2}e^{-2c_0}J_L(\alpha) - \frac{1}{8N}e^{-2c_0}\left(\sum_{i=1}^N \|\widehat{y}\,(x_i,\alpha) - y_i\|_2\right)^2\right) \\
&\geq \frac{1}{LN^2}\left(\frac{N}{2}e^{-2c_0}J_L(\alpha) - \frac{N}{4}e^{-2c_0}J_L(\alpha)\right) = \frac{1}{4N}e^{-2c_0}L^{-1}J_L(\alpha).
\end{aligned}$$

□

Next, if we assume that the weights $\alpha^{(L)}$ are close to each other in neighbouring layers, we can deduce that the gradient of the loss with respect to weights in neighbouring layers are also close to each other. Hence, if we couple this fact with Lemma 4.2, we can prove a lower bound on the norm of the gradient of the loss with respect to the full weight vector $\alpha^{(L)}$.

Lemma 4.3. *Under Assumption 4.1 (i)–(iii), let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ such that $L \geq \max(5c_0, 4c_0^2)$, $\|\alpha^{(L)}\|_{F,\infty} \leq c_0 L^{-1/2}$, and $\left\|\alpha_{k+1}^{(L)} - \alpha_k^{(L)}\right\|_F \leq 2^{-7/2}N^{-1/2}e^{-4.2c_0}L^{-1}$ for each k. Then,*

$$\left\|\nabla_{\alpha^{(L)}} J_L\big(\alpha^{(L)}\big)\right\|_F^2 \geq \left(\frac{1}{16}N^{-1}e^{-2c_0} - 17dc_0^4e^{6.4c_0}L^{-1}\right)J_L(\alpha).$$

Proof. Fix $L \geq \max(5c_0, 4c_0^2)$. In the proof, we omit the explicit dependence in L. We use Lemma B.5 to estimate the difference of neighbouring gradients:

$$\begin{aligned}
\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}} &= \frac{\delta_L}{N}\sum_{i=1}^N h_{k-1,n}^{x_i}\left(\dot{\sigma}_{k,x_i,m} - \dot{\sigma}_{k+1,x_i,m}\right)\nabla_{\widehat{y}}\,\ell\left(y_i, \widehat{y}(x_i,\alpha)\right)^\top M_{k+1}^{x_i}e_m \\
&\quad + \frac{\delta_L^2}{N}\sum_{i=1}^N \nabla_{\widehat{y}}\,\ell\left(y_i, \widehat{y}(x_i,\alpha)\right)^\top M_{k+1}^{x_i}\xi_{k,mn}^{x_i,(L)},
\end{aligned}$$

where $\xi^{x,(L)}_{k,mn}$ satisfies

$$\left\|\xi^{x,(L)}_{k,mn}\right\|_2^2 \leq 2\left(h^x_{k-1,n}\right)^2 \left\|\alpha_{k+1}-\alpha_k\right\|_F^2 + 2\left\|\alpha_{k,n}\right\|_2^4 \left\|h^x_{k-1}\right\|_2^4.$$

By Lemma B.1 and the fact that σ' is $1-$Lipschitz by Assumption 4.1 *(i)*, we bound further:

$$\begin{aligned}
\left\|\nabla_{\alpha_{k+1}} J_L\left(\alpha\right) - \nabla_{\alpha_k} J_L\left(\alpha\right)\right\|_F^2 &= \sum_{m,n=1}^{d}\left(\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}}\right)^2 \\
&\leq 4\sum_{m,n=1}^{d}\frac{1}{LN}\sum_{i=1}^{N}\left(h^{x_i}_{k-1,n}\right)^2 \left\|M^{x_i}_k e_m\right\|_2^2 \left(\alpha_k h^{x_i}_{k-1} - \alpha_{k+1}h^{x_i}_k\right)_m^2 \ell(y_i,\widehat{y}(x_i,\alpha)) \\
&\quad + \frac{4}{L^2N}\sum_{i=1}^{N} e^{2c_0}\left(2de^{2.2c_0}\left\|\alpha_{k+1}-\alpha_k\right\|_F^2 + 2dc_0^4e^{4.4c_0}L^{-2}\right)\ell(y_i,\widehat{y}(x_i,\alpha)) \\
&\leq e^{4.2c_0}\frac{4}{LN}\sum_{i=1}^{N}\left\|\alpha_k h^{x_i}_{k-1} - \alpha_{k+1}h^{x_i}_k\right\|_2^2 \ell(y_i,\widehat{y}(x_i,\alpha)) + 9dc_0^4e^{6.4c_0}L^{-4}J_L(\alpha).
\end{aligned}$$

Then, simply note that

$$\begin{aligned}
\left\|\alpha_k h^{x_i}_{k-1} - \alpha_{k+1}h^{x_i}_k\right\|_2^2 &\leq 2\left\|\left(\alpha_{k+1}-\alpha_k\right)h^{x_i}_k\right\|_2^2 + 2\left\|\alpha_k\left(h^{x_i}_k - h^{x_i}_{k-1}\right)\right\|_2^2 \\
&\leq \frac{1}{64}N^{-1}e^{-6.2c_0}L^{-2} + 2c_0^4e^{2.2c_0}L^{-3}.
\end{aligned}$$

Hence,

$$\left\|\nabla_{\alpha_{k+1}} J_L\left(\alpha\right) - \nabla_{\alpha_k} J_L\left(\alpha\right)\right\|_F^2 \leq \left(\frac{1}{16}N^{-1}e^{-2c_0} + 17dc_0^4e^{6.4c_0}L^{-1}\right)L^{-3}J_L(\alpha).$$

Finally, we use the reverse triangle inequality and Cauchy-Schwarz inequality:

$$\begin{aligned}
\left\|\nabla_{\alpha_k} J_L\left(\alpha\right)\right\|_F^2 &\geq \frac{1}{2}\left\|\nabla_{\alpha_1} J_L\left(\alpha\right)\right\|_F^2 - (k-1)\sum_{k'=1}^{k-1}\left\|\nabla_{\alpha_{k+1}} J_L\left(\alpha\right) - \nabla_{\alpha_k} J_L\left(\alpha\right)\right\|_F^2 \\
&\geq \frac{1}{8}N^{-1}e^{-2c_0}L^{-1}J_L(\alpha) - \frac{(k-1)^2}{L^3}\left(\frac{1}{16}N^{-1}e^{-2c_0} + 17dc_0^4e^{6.4c_0}L^{-1}\right)J_L(\alpha) \\
&\geq \left(\frac{1}{16}N^{-1}e^{-2c_0} - 17dc_0^4e^{6.4c_0}L^{-1}\right)L^{-1}J_L(\alpha).
\end{aligned}$$

The second inequality holds by Lemma 4.2 and *(i)* above. Hence,

$$\left\|\nabla_\alpha J_L\left(\alpha^{(L)}\right)\right\|_F^2 = \sum_{k=1}^{L}\left\|\nabla_{\alpha_k} J_L\left(\alpha^{(L)}\right)\right\|_F^2 \geq \left(\frac{1}{16}N^{-1}e^{-2c_0} - 17dc_0^4e^{6.4c_0}L^{-1}\right)J_L(\alpha).$$

□

It guarantees that for $L \gg 1$, every critical point close to the origin is a global minimum of the objective function, similarly to what is known for linear residual networks [68, 87, 93, 104].

4.3.2 Behaviour of weight norms along the gradient descent path

In Section 4.3.1, we establish bounds on the gradient of the loss function evaluated at a generic weight vector $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$. We now proceed to understand how $\alpha^{(L)}$ changes under a gradient descent update. To do so, we study the local version of the weight norms defined in (4.5). Define for $x, y \in \mathbb{R}^d$ and $k = 0, \dots, L$:

$$G_k^{x,y,(L)}\big(\alpha^{(L)}\big) := \frac{\partial \ell(y,\,\cdot\,)}{\partial h_k^{(L)}}\left(\widehat{y}\big(x, \alpha^{(L)}\big)\right) \in \mathbb{R}^d. \tag{4.9}$$

Also, for clarity, denote $h_k^{x,(L)}\big(\alpha^{(L)}\big) \in \mathbb{R}^d$ for the hidden state of the k^{th} layer using input $x \in \mathbb{R}^d$ and network weights $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$.

Lemma 4.4. *Let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ and define $\widetilde{\alpha}^{(L)} := \alpha^{(L)} - \eta_L \nabla_\alpha J_L\left(\alpha^{(L)}\right)$. Define further $f_{k,m}^{(L)}\left(\alpha^{(L)}\right) := \frac{1}{2}L\left\|\alpha_{k,m}^{(L)}\right\|_2^2$ Under Assumption 4.1 (i)–(ii), we have*

$$f_{k,m}^{(L)}\left(\widetilde{\alpha}^{(L)}\right)^{1/2} \le f_{k,m}^{(L)}\left(\alpha^{(L)}\right)^{1/2} + \frac{1}{\sqrt{2}}\eta_L\left(\frac{1}{N}\sum_{i=1}^{N}\left\|h_{k-1}^{x_i,(L)}\big(\alpha^{(L)}\big)\right\|_2^2\left\|G_k^{x_i,y_i,(L)}\big(\alpha^{(L)}\big)\right\|_\infty^2\right)^{1/2}.$$

Lemma 4.5. *Let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ and $c_A > 0$ such that $L \ge 5c_A$ and $\left\|\alpha^{(L)}\right\|_{F,\infty} \le c_A L^{-1/2}$. Define $\widetilde{\alpha}^{(L)} := \alpha^{(L)} - \eta_L \nabla_\alpha J_L\left(\alpha^{(L)}\right)$, and let $g_k^{(L)}\left(\alpha^{(L)}\right) := \frac{1}{2}L^2\left\|\alpha_{k+1}^{(L)} - \alpha_k^{(L)}\right\|_F^2$. Under Assumption 4.1 (i)–(ii), we have*

$$\begin{aligned} g_k^{(L)}\left(\widetilde{\alpha}^{(L)}\right) \le g_k^{(L)}\left(\alpha^{(L)}\right)&\left(1 + L^{-1/2}\eta_L\frac{1}{N}\sum_{i=1}^{N}\left\|h_{k-1}^{x_i,(L)}\left(\alpha^{(L)}\right)\right\|_2^2\left\|G_{k+1}^{x_i,y_i,(L)}\left(\alpha^{(L)}\right)\right\|_\infty\right)^2 \\ &+ \mathcal{O}\Big(c_A e^{2.1c_A}\left(c_A e^{1.1c_A}L^{-1} + 2L^{-1/2}\right)\eta_L g_k\big(\alpha^{(L)}\big)^{1/2} J_L\big(\alpha^{(L)}\big)^{1/2}\Big), \end{aligned}$$

where the Big-O constant is also independent of c_A.

The proofs of Lemmas 4.4 and 4.5 can be found in Appendix B.5.

4.3.3 Local convergence of gradient descent

In this section, we initialize the weight vector $A^{(L)}(0)$ according to Assumption 4.1 *(iv)* and we let the weights $A^{(L)}(t)$ evolve according to the gradient descent dynamics (4.3). We show that under some *a priori* conditions on the initial parameters, the initial loss, and the learning rates, we are able to prove a practical upper bound on the loss function along the gradient descent path.

Theorem 4.6. *Let L be large enough. Under Assumption 4.1 (i)–(v), let the parameter $A^{(L)}(t)$ evolve according to the gradient descent dynamics (4.3) with learning rates $\eta_L(t)$ until time $T_L \in \mathbb{N}$, chosen in such a way that for each $t = 0, \dots, T_L - 1$, we have*

$$\eta_L(t) \leq \frac{1}{160} N^{-1} d^{-1} e^{-10.5c_0} \quad \textit{and} \quad \sum_{t=0}^{T_L-1} \eta_L(t) \leq d^{-1} \log L. \tag{4.10}$$

Then, for each $t = 0, \dots, T_L$, we have

$$J_L(A(t)) \leq \exp\left(-\frac{1}{32} N^{-1} e^{-2c_0} \sum_{t'=0}^{t-1} \eta_L(t')\right) J_0 + 34 d c_0^4 e^{6.4c_0} \left(\sum_{t'=0}^{t-1} \eta_L(t')\right) L^{-1} J_0.$$

Theorem 4.6 is a local convergence result since we assume that the initial loss lies below a certain level by Assumption 4.1 *(v)*. We are able to show convergence as $L \to \infty$ of the loss to zero when the horizon T_L depends explicitly on the depth while satisfying (4.10).

Proof. We choose L big enough so that

$$\frac{3}{64} N^{-1} d^{-1} c_0^2 e^{2.2c_0} (\log L)^{3/2} \leq L^{1/2}, \quad 34 c_0^4 e^{6.4c_0} \log L \leq L. \tag{4.11}$$

Note that it trivially implies that $L \geq \max(4c_0^2, 5c_0)$. In the proof, we omit the explicit dependence in L. Denote $J_0 := J_L(A(0))$ the initial loss. We first prove jointly that

$$\begin{aligned} J_L(A(t)) &\leq 2J_0, \\ \max_k \|A_k(t)\|_F &\leq c_0 L^{-1/2}, \\ \max_k \|A_{k+1}(t) - A_k(t)\|_F &\leq 2^{-7/2} N^{-1/2} e^{-4.2c_0} L^{-1}. \end{aligned} \tag{4.12}$$

for $t = 0, \dots, T_L$ by induction on t. For $t = 0$, by Assumption 4.1 *(iv)*, we directly have

$$\begin{aligned} \max_k \|A_k(0)\|_F &\leq d^{1/2} \sup_{k,m} \|A_{k,m}(0)\|_2 \leq L^{-1} < c_0 L^{-1/2}, \\ \|A_{k+1}(0) - A_k(0)\|_F &\leq d^{1/2} \sup_m \|A_{k+1,m}(0) - A_{k,m}(0)\|_2 < 2^{-7/2} N^{-1/2} e^{-4.2c_0} L^{-1}. \end{aligned} \tag{4.13}$$

Let $t \geq 0$. Assume that (4.12) holds true for all $t' \leq t < T_L$. We prove that (4.12) holds for $t+1$. Define $f_{k,m}(t) := f_{k,m}\left(A^{(L)}(t)\right)$ as in Lemma 4.4 and $g_k(t) := g_k\left(A^{(L)}(t)\right)$ as in Lemma 4.5. As $L \geq \max(4c_0^2, 5c_0)$, we can apply Lemma 4.4 and Lemma B.1 with the induction hypothesis.

$$\begin{aligned} f_{k,m}(t+1)^{1/2} &\leq f_{k,m}(t)^{1/2} + \frac{1}{\sqrt{2}} e^{2.1c_0} \eta_L(t) \left(\frac{2}{N} \sum_{i=1}^N \ell\left(y_i, \widehat{y}(x_i, A(t))\right)\right)^{1/2} \\ &= f_{k,m}(t)^{1/2} + e^{2.1c_0} \eta_L(t) J_L(A(t))^{1/2}. \end{aligned} \tag{4.14}$$

Similarly, we apply Lemma 4.5 with $c_A = c_0$ and Lemma B.1 with the induction hypothesis.

$$\begin{aligned} g_k(t+1) \le g_k(t) \left(1 + e^{3.2c_0} \eta_L(t) L^{-1/2} J_L(A(t))^{1/2}\right)^2 \\ + \mathcal{O}\Big(c_0 e^{2.1c_0} \left(c_0 e^{1.1c_0} L^{-1} + 2L^{-1/2}\right) \eta_L g_k(t)^{1/2} J_L\big(A(t)\big)^{1/2}\Big). \end{aligned} \tag{4.15}$$

Now, we want to apply Lemma B.6 to bound $J_L(A(t))$. We check that using Lemma 4.3, the assumptions of Lemma B.6 are verified for

$$c_A(t') = c_0, \quad \underline{c} \equiv \underline{c}(t') = \frac{1}{16} N^{-1} e^{-2c_0}, \quad \bar{c} \equiv \bar{c}(t') = 34 d c_0^4 e^{6.4c_0} J_0 \quad \text{for } t' \le t.$$

Thus, as $\eta_L(t) < 2^{-5} 5^{-1} N^{-1} d^{-1} e^{-10.5c_0} < 2^{-1} c_0 e^{-3.2c_0}$, we deduce the following bound on the loss function at all times $t' = 0, \dots, t+1$.

$$J_L(A(t')) \le \exp\left(-\frac{1}{2}\underline{c} \sum_{t''=0}^{t'-1} \eta_L(t'')\right) J_0 + \bar{c} L^{-1} \sum_{t''=0}^{t'-1} \eta_L(t'') \tag{4.16}$$

Bound on $J_L(A(t+1))$: Plugging in (4.10) and (4.11) into (4.16), we verify that

$$J_L(A(t+1)) \le \left(1 + 34 c_0^4 e^{6.4c_0} L^{-1} \log L\right) J_0 \le 2 J_L(A(0)).$$

Bound on $f_{k,m}(t+1)$: We plug (4.16) into (4.14) and sum over t to deduce

$$\begin{aligned} f_{k,m}(t+1)^{1/2} &\le f_{k,m}(0)^{1/2} + e^{2.1c_0} \sum_{t'=0}^{t} \eta_L(t') J_L(A(t'))^{1/2} \\ &\le \frac{1}{3\sqrt{2}} d^{-1/2} c_0 L^{-1/2} + e^{2.1c_0} R_L(t), \end{aligned} \tag{4.17}$$

where we use (4.13) for the second inequality and

$$R_L(t) \coloneqq \sum_{t'=0}^{t} \eta_L(t') J_L(A(t'))^{1/2}.$$

To find an upper bound to $R_L(t)$, we use the inequality $\sqrt{x+y} \le \sqrt{x} + \sqrt{y}$ in (4.16), with the help of (4.11):

$$R_L(t) \le \sum_{t'=0}^{t} \eta_L(t') \exp\left(-\frac{1}{4}\underline{c} \sum_{t''=0}^{t'-1} \eta_L(t'')\right) J_0^{1/2} + \bar{c}^{1/2} L^{-1/2} \sum_{t'=0}^{t} \eta_L(t') \left(\sum_{t''=0}^{t'-1} \eta_L(t'')\right)^{1/2}$$

Now, we estimate the following quantity using (4.10):

$$\sum_{t'=0}^{t} \eta_L(t') \left(\sum_{t''=0}^{t'-1} \eta_L(t'')\right)^{1/2} \le \left(\sum_{t'=0}^{t} \eta_L(t')\right)^{3/2} \le d^{-3/2} (\log L)^{3/2}.$$

Next, we use the fact $\eta_L(t) < \bar{\eta} = 2^{-5}5^{-1}N^{-1}d^{-1}e^{-10.5c_0}$ and

$$\left(1 - \exp\left(-\frac{1}{4}\underline{c}\bar{\eta}\right)\right) x \leq \bar{\eta}\left(1 - \exp\left(-\frac{1}{4}\underline{c}x\right)\right) \quad \text{for all } x \in [0, \bar{\eta}]$$

to deduce that the following sum is telescoping:

$$\begin{aligned}\sum_{t'=0}^{t} \eta_L(t') \exp\left(-\frac{1}{4}\underline{c}\sum_{t''=0}^{t'-1} \eta_L(t'')\right) &\leq \frac{1 - \exp\left(-\frac{1}{4}\underline{c}\sum_{t'=0}^{t} \eta_L(t')\right)}{1 - \exp(-\frac{1}{4}\underline{c}\bar{\eta})}\bar{\eta} \\ &\leq 8\underline{c}^{-1}.\end{aligned}$$

Hence, by Assumption 4.1 *(v)* and (4.11),

$$\begin{aligned}R_L(t) &\leq 128Ne^{2c_0}J_0^{1/2} + 6d^{1/2}c_0^2e^{3.2c_0}(\log L)^{3/2}L^{-1/2}J_0^{1/2} \\ &\leq \frac{2c_0}{3\sqrt{2}}d^{-1/2}e^{-2.1c_0}\end{aligned} \tag{4.18}$$

Plugging it in (4.17), we obtain

$$f_{k,m}(t+1)^{1/2} \leq \frac{c_0}{3\sqrt{2}}d^{-1/2} + \frac{2c_0}{3\sqrt{2}}d^{-1/2} = \frac{c_0}{\sqrt{2}}d^{-1/2}.$$

Hence, this completes the induction step for the norm of A:

$$\left\|A^{(L)}(t+1)\right\|_{F,\infty} \leq \sqrt{2}d^{1/2}L^{-1/2}\sup_{k,m} f_{k,m}(t+1)^{1/2} \leq c_0L^{-1/2}.$$

Bound on $g_k(t+1)$: By (4.11), $L^{1/2} \geq c_0e^{1.1c_0}$, so we can rewrite (4.15):

$$g_k(t+1) \leq g_k(t)u_L(t) + g_k(t)^{1/2}\,\mathcal{O}\left(c_0e^{2.1c_0}\eta_L(t)L^{-1/2}J_L(A(t))^{1/2}\right),$$

where

$$u_L(t) := \left(1 + \eta_L(t)L^{-1/2}e^{3.2c_0}J_L(A(t))^{1/2}\right)^2.$$

We can thus apply Lemma D.4 *(ii)*, together with the identity $1 + x \leq \exp(x)$ and (4.18) to deduce that

$$\begin{aligned}g_k(t+1)^{1/2} &\leq \exp\left(e^{3.2c_0}R_L(t)L^{-1/2}\right)\left(g_k(0)^{1/2} + \mathcal{O}\left(c_0e^{2.1c_0}R_L(t)L^{-1/2}\right)\right) \\ &\leq \exp\left(c_0e^{1.1c_0}L^{-1/2}\right)\left(1 + \mathcal{O}(c_0^2L^{-1/2})\right)g_k(0)^{1/2} \\ &\leq \sqrt{2}g_k(0)^{1/2}.\end{aligned}$$

The last inequality is derived with the help of (4.11). We finish the induction step by observing that $g_{k,m}(0)^{1/2} \leq 2^{-9/2}N^{-1/2}d^{-1/2}e^{-4.2c_0}$ by Assumption 4.1 *(iv)*.

Convergence of $J_L(A(T_L)) \to 0$: We now have all the tools to deduce the rate of convergence of $J_L(A(T_L))$ to zero. We observe from the induction result above that the assumptions of Lemma B.6 are verified for $c_A(t) = c_0$, $\underline{c}(t) = 2^{-4}N^{-1}e^{-2c_0}$ and $\bar{c}(t) = 34dc_0^4 e^{6.4c_0} J_0$ by Lemma 4.3, for each $t \in [0, T)$. In particular, we have

$$J_L(A(T_L)) \leq \exp\left(-\frac{1}{32}N^{-1}e^{-2c_0}\sum_{t=0}^{T_L-1}\eta_L(t)\right)J_0 + 34dc_0^4e^{6.4c_0}\left(\sum_{t=0}^{T_L-1}\eta_L(t)\right)L^{-1}J_0. \tag{4.19}$$

□

Remark 4.7. *Let $\eta_0 > 0$ be a fixed learning rate, independent of k, L and t, and let $J_0 := J_L\big(A^{(L)}(0)\big)$ be the initial loss. Observe that from Theorem 4.6, if we choose*

- $\eta_L(t) = \eta_0$ *and* $T_L^{\text{const}} = \Theta(\eta_0^{-1}\log L)$, *then conditions (4.10) are satisfied, so we deduce*

  $$J_L\big(A^{(L)}(T_L^{\text{const}})\big) \leq \exp(-\underline{c}\eta_0 T_L^{\text{const}})J_0 + \mathcal{O}(\eta_0 T_L^{\text{const}}L^{-1}).$$

  Hence, for an error level $\epsilon > 0$, gradient descent with constant learning rate for a network of depth $L = \Omega(1/\epsilon)$ reaches $J_L\big(A^{(L)}(T_L^{\text{const}})\big) < \epsilon$ in $\Theta(\eta_0^{-1}\log 1/\epsilon)$ iterations.

- $\eta_L(t) = \eta_0(t+1)^{-1}$ *and* $T_L^{\text{decay}} = \Theta(\exp(\eta_0^{-1}\log L))$, *then conditions (4.10) are satisfied. We deduce that*

  $$J_L\Big(A^{(L)}(T_L^{\text{decay}})\Big) \leq \exp(-\underline{c}\eta_0 \log T_L^{\text{decay}})J_0 + \mathcal{O}(\eta_0 \log T_L^{\text{decay}} L^{-1}).$$

  Hence, for an error level $\epsilon > 0$, gradient descent with decaying learning rate for a network of depth $L = \Omega(1/\epsilon)$ reaches $J_L\Big(A^{(L)}(T_L^{\text{decay}})\Big) < \epsilon$ in $\Theta(\exp(\eta_0^{-1}\log 1/\epsilon))$ iterations.

The above convergence rates above are confirmed by our experiments in Section 4.4. Note that gradient descent converges exponentially faster when using constant learning rates rather than decaying ones. This is because the parameters $A^{(L)}(t)$ and the gradients $\nabla_A J_L(A^{(L)}(t))$ are already on the same scale $\mathcal{O}(L^{-1/2})$. Note also that Theorem 4.6 is not in contradiction with [16, Theorem 6] stating that gradient descent might get stuck at the critical point $\big(\delta^{(L)}, A^{(L)}\big) = (0,0)$ that is usually not a global minimizer. Indeed, we force $\delta^{(L)}$ to have a non-trivial scaling by Assumption 4.1 *(iv)*, so that $(0,0)$ is simply not a point in the parameter space.

4.3.4 Scaling limit of trained weights

In many cases the trained weights, viewed as a function of the layer index k/L, have a scaling limit which is a function defined on $[0,1]$. We show that such a limit then admits finite p-variation with $p=2$.

Proposition 4.8. *Let $\left(A^{(L)}(t)\colon t=1,\ldots,T_L\right)$ follow the gradient descent dynamics (4.3), where the assumptions of Theorem 4.6 are satisfied for $T=T_L$. Assume there exists $\overline{A}^* := [0,1] \to \mathbb{R}^{d\times d}$ such that*

$$\sup_{s\in[0,1]} L^{1/2} \left\| L^{1/2} A^{(L)}_{\lfloor Ls \rfloor}(T_L) - \overline{A}^*_s \right\|_F \xrightarrow{L\to\infty} 0 \tag{4.20}$$

Then, the scaling limit $\overline{A}^*$ has finite p-variation with $p=2$.

Conditions 4.20 may seem strong, but they are related to the norm $f^{(L)}_{k,m}(T_L)$ defined in Lemma 4.4 having a limit as $L\to\infty$. Under the hypothesis of Theorem 4.6, we have shown in the proof of Theorem 4.6 that the norm $f^{(L)}_{\lfloor Ls\rfloor,m}(T_L)$ stay uniformly bounded (in s and m) as $L\to\infty$. Condition 4.20 has also been verified in numerical experiments, see Section 4.4.3.

Proof. Fix a partition $\pi = \{0 = s_0 < s_1 < \ldots < s_K = 1\}$, where the mesh of the partition $\|\pi\|$ is small enough. In the following, $c>0$ denotes a constant independant of s and L. For $i=1,\ldots,K-1$, let $L_i\in\mathbb{N}$ big enough so that Theorem 4.6 applies. We estimate directly

$$\begin{aligned}\left\|\overline{A}^*_{s_{i+1}} - \overline{A}^*_{s_i}\right\|_F &\le \left\| L_i^{1/2} A^{(L_i)}_{\lfloor L_i s_{i+1}\rfloor}(T_{L_i}) - \overline{A}^*_{s_{i+1}}\right\|_F + \left\| L_i^{1/2} A^{(L_i)}_{\lfloor L_i s_i\rfloor}(T_{L_i}) - \overline{A}^*_{s_i}\right\|_F \\ &\quad + L_i^{1/2}\left\| A^{(L_i)}_{\lfloor L_i s_{i+1}\rfloor}(T_{L_i}) - A^{(L_i)}_{\lfloor L_i s_i\rfloor}(T_{L_i})\right\|_F \\ &\le cL_i^{-1/2} + L_i^{1/2}\left\| A^{(L_i)}_{\lfloor L_i s_{i+1}\rfloor}(T_{L_i}) - A^{(L_i)}_{\lfloor L_i s_i\rfloor}(T_{L_i})\right\|_F\end{aligned}$$

We now use the proof of Theorem 4.6 to deduce a uniform bound (in k and L) on the quantity $g^{(L)}_k(T_L)$ defined in Lemma 4.5. That means, $L\left\|A^{(L)}_{\lfloor Ls\rfloor}(T_L) - A^{(L)}_{\lfloor Ls\rfloor+1}(T_L)\right\|_F < c < \infty$. We can apply the triangle inequality to deduce

$$\left\|\overline{A}^*_{s_{i+1}} - \overline{A}^*_{s_i}\right\|_F \le cL_i^{-1/2} + cL_i^{-1/2}\left(\lfloor L_i s_{i+1}\rfloor - \lfloor L_i s_i\rfloor\right) \le cL_i^{-1/2} + cL_i^{1/2}\left|s_{i+1}-s_i\right|.$$

Hence,

$$\sum_{i=0}^{K-1}\left\|\overline{A}^*_{s_{i+1}} - \overline{A}^*_{s_i}\right\|_F^2 \le c\sum_{i=0}^{K-1} L_i^{-1} + L_i\left|s_{i+1}-s_i\right|^2. \tag{4.21}$$

As $\|\pi\|$ is small enough, we can choose $L_i = \Theta(|s_{i+1}-s_i|^{-1})$ to deduce that the RHS of (4.21) is bounded uniformly in π. Taking a supremum over all such partitions then show that $\overline{A}^*$ has finite p-variation with $p=2$. □

4.4 Numerical experiments

To illustrate the results of Section 4.3, we design numerical experiments with the following set-up. We have a fixed training set $\{(x_i, y_i) : i = 1, \dots, N\}$ in $\mathbb{R}^d \times \mathbb{R}^d$, where d is the dimension of the inputs and outputs and N is the size of the dataset. For any depth $L \in \mathbb{N}$, we initialize the weights of the network (4.1) with $\delta_L = L^{-\alpha_0}$ and each entry of $A_k^{(L)}$ is independent and normally distributed with standard deviation $d^{-1}L^{-\beta_0}$, where $\alpha_0, \beta_0 \in [0, 1]$. The weights are trained using gradient descent on the (unregularized) mean squared error J_L defined in (4.2) with a fixed learning rate η_0 independent of d, k, L and the training time t. We perform a fixed number $T \in \mathbb{N}$ of gradient updates, with no early stopping.

4.4.1 Identification of scaling behavior

We run two experiments to discover the best scaling for δ_L. Denote α_t the scaling of δ_L at time t, i.e. $\alpha_t \propto L^{-\alpha_t}$, and denote β_t the scaling of the weights $A^{(L)}(t)$ at time t, i.e. $A^{(L)}(t) \propto L^{-\beta_t}$. The first experiment is to let δ_L trainable with gradient descent with learning rate η_0, and observe the resulting scaling α_t. We observe in Figure 4.1 (left) that α_t tend to get closer to $1/2$ as t increases. However, this is far from being exact, even though the networks have all converged, see Figure 4.1 (right). It is interesting to note that $\alpha_0 = 1/2$ is a *fixed point*, meaning that the networks initialized with this scaling will keep $\alpha_t \approx 1/2$ during the entire training. The second experiment is to let $\delta_L = L^{-\alpha_0}$ at initialization and keep it fixed during training, i.e. $\alpha_t = \alpha_0$ for each t. We thus have weights $A^{(L)}(0)$ that scale like $L^{-\beta_0}$ initially, and that are updated with $\eta_L(t)\nabla_{A_k} J_L(A^{(L)}(t)) \propto L^{-\alpha_0} J_L(A^{(L)}(t))^{1/2}$ by Lemma B.3. Thus, it is

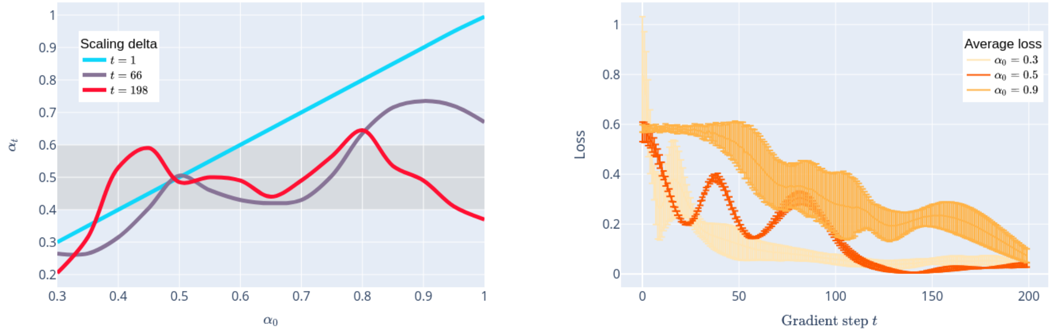

Figure 4.1: Left: scaling α_t of δ_L against the initial scaling α_0 for different training times. Right: Average loss value across depths $L \in \{2^k : k \in [3, 12]\}$ for different initializations α_0, as a function of the number of gradient steps t.

reasonable to expect that if $\beta_0 > \alpha_0$, and the loss J_L at small times t is independent of the depth, then $\beta_t \approx \alpha_0$ for small times t. In fact, we observe in Figure 4.2 (left)

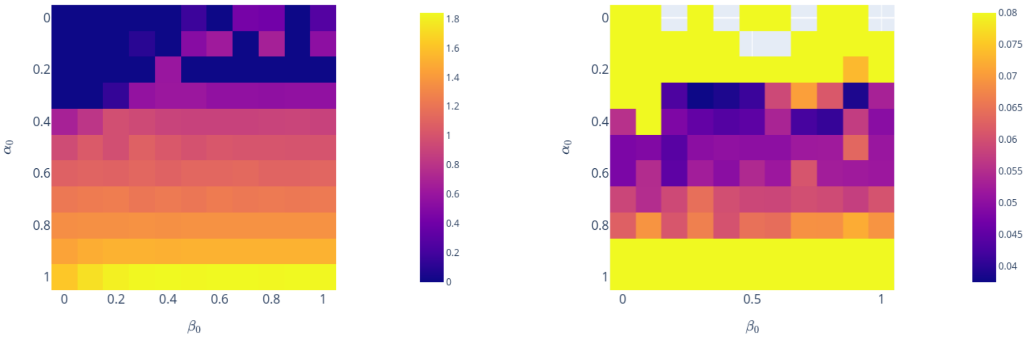

Figure 4.2: Both figures: horizontal axis is the initial scaling β_0 of the weights A, and the vertical axis is the fixed scaling α_0 of δ_L. Left: Final total scaling $\alpha_0 + \beta_T$. Right: Average final loss after $T = 200$ epochs. The depths at which we train our networks are $L \in \left\{2^k : k \in [3, 10]\right\}$.

that the total scaling $\alpha_0 + \beta_T$ is independent of β_0 and is roughly equal to $2\alpha_0$. We observe in Figure 4.2 (right) that the parameters that gives the best performance is around $\alpha_0 = 1/2$, again independently of β_0. This is expected, as

$$h_k^{(L)} - h_{k-1}^{(L)} = \delta_L \sigma_d \left(A_k^{(L)} h_{k-1}^{(L)}\right) \propto L^{-\alpha_0 - \beta},$$

so the final scaling of the increments of the hidden states is roughly $2\alpha_0$, which should be around 1 to guarantee stability of the large depth limit.

4.4.2 Rate of convergence

We now verify that the convergence rates of gradient descent agree with the theoretical rates derived in Remark 4.7. To do so, we run our experiments with different initial learning rates, and take the average loss curve across the depths. We then plot the number of gradient steps needed to reach a certain loss level. We observe in Figure 4.3 that the number of gradient steps needed to attain a given level ϵ is linear in $\log(1/\epsilon)$ for constant learning rates, and exponential in $\log(1/\epsilon)$ for learning rates decaying like $1/t$. We also see that in both cases, the rate of convergence is inversely proportional to the initial learning rate η_0.

4.4.3 Emergence of regularity of weights as a function of the layer index

Recall the results of Proposition 4.8 stating that under condition (4.20), the rescaled trained weights $L^{1/2} A_{\lfloor Ls \rfloor}^{(L)}(T)$ converge to a limit $\overline{A}_s^*$ that has finite 2-variation. We

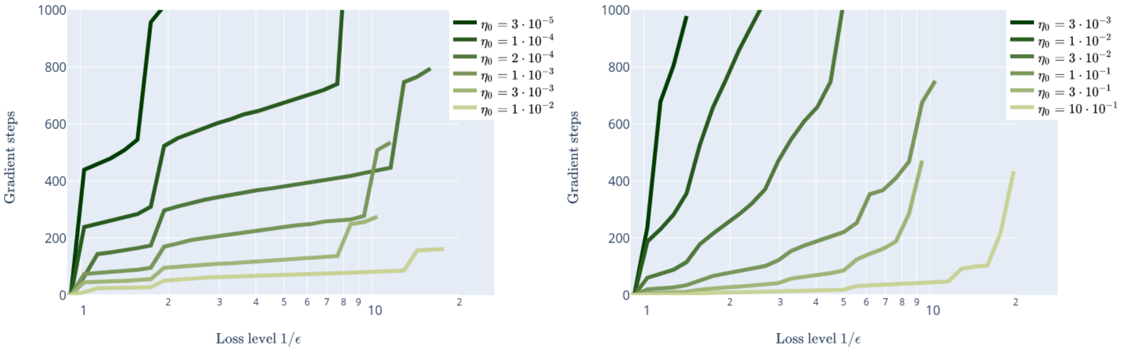

Figure 4.3: Both figures: horizontal axis is the inverse loss level $1/\epsilon$, in log-scale, and the vertical axis is the number of gradient steps needed for the average loss to drop below ϵ. The average is taken over the depths $L \in \{2^k : k \in [3, 10]\}$. Left: constant learning rates $\eta_L(t) = \eta_0$. Right: decaying learning rates $\eta_L(t) = \eta_0(t+1)^{-1}$.

verify that condition (4.20) holds by running experiments for varying depths and looking at the quantities

$$\overline{f}^{(L)}(t) := \frac{1}{2}\sum_{k=1}^{L}\left\|A_k^{(L)}(t)\right\|_F^2 \quad \text{and} \quad \overline{g}^{(L)}(t) := \frac{1}{2}L\sum_{k=1}^{L-1}\left\|A_{k+1}^{(L)}(t) - A_k^{(L)}(t)\right\|_F^2.$$

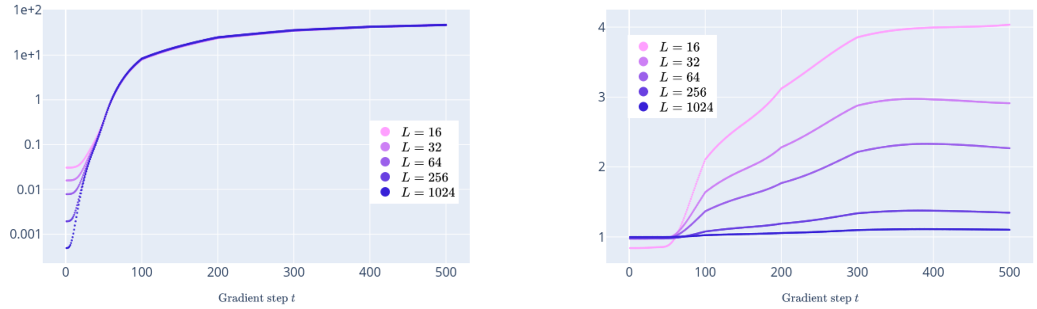

Figure 4.4: Evolution of weight norms along gradient descent path for different depths $L \in \{2^4, 2^5, 2^6, 2^8, 2^{10}\}$. Left: L^2-type norm $\overline{f}^{(L)}(t)$ as a function of gradient iterations. Right: Quadratic variation-type norm $\overline{g}^{(L)}(t)$ as a function of gradient iterations.

We observe in Figure 4.4 that at initialization $t = 0$, the sum of the squared norms $\overline{f}^{(L)}$ is $\mathcal{O}(L^{-1})$, and becomes $\mathcal{O}(1)$ during training $t \gg 1$. However, the smoothness of the weights as measured by $\overline{g}^{(L)}(t)$ is constant with t for large L. That means, the conservation of smoothness during training is a feature of the architecture (smooth activation function) and of gradient descent, not of the particular weight initialization nor of a particular scaling.

We observe in Figure 4.5 that as $L \to \infty$, the rescaled trained weights converge to a limit $\overline{A}^*$. This is a striking result, indicative of the stability of this network architecture [67]: there is no *a priori* reason that networks with different depths and trained independently of each other should behave similarly. The limiting behaviour of trained weights of residual networks with a smooth activation function was first observed in [36], where the limit is explicitly derived and proved.

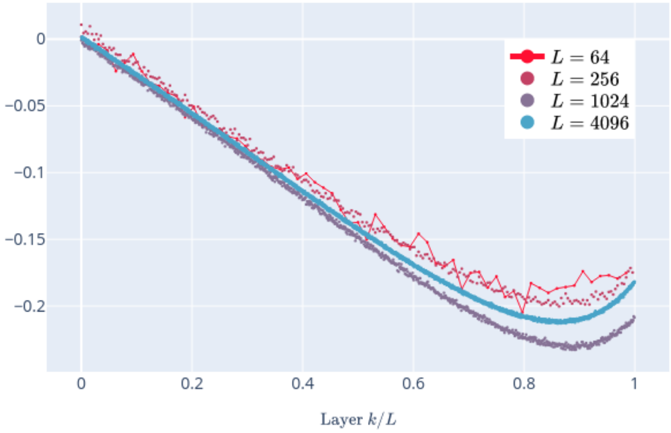

Figure 4.5: Scatter plot of the rescaled weights $L^{1/2}A^{(L)}_{k,(7,18)}(T)$ for different values of $L \in \{4^x : x \in [3, 6]\}$ at the end of the training $T = 500$. Horizontal axis is the scaled layer index k/L.

4.5 Conclusion

We prove linear convergence of gradient descent to a global minimum of the training loss for deep residual networks with constant layer width and smooth activation function. We further show that if the trained weights, as a function of the layer index, admits a scaling limit as the depth of the network tends to infinity, then it has finite 2−variation.

A natural question to investigate next is the generalization capability of the trained weights obtained by gradient descent, which we characterize in this work. Indeed, it is still an open question whether the weights obtained by gradient descent admit the tightest generalization gap among all the other global minima. Also, our work can be generalized to study other residual architectures (for example with ReLU activation) by looking at alternative norms along the gradient descent path.

Chapter 5

Mean-field limit and global convergence of gradient descent for path-homogeneous models

5.1 Introduction

Many tasks in machine learning, including random feature selection [112, 130, 134], matrix factorization [8, 66], ensemble averaging [121, 137] and training a two layer neural network [31, 114, 133, 141] can be formulated as a minimization of a smooth convex functional of a positive measure

$$\min_{\mu \in \mathcal{M}_+(\Theta)} F(\mu) = L\left(\int_\Theta \Phi(\theta) \mathrm{d}\mu(\theta)\right) + \int_\Theta V(\theta) \mathrm{d}\mu(\theta), \tag{5.1}$$

where $\Phi \colon \Theta \to \mathcal{H}$ is a smooth function from the set of parameters $\Theta \subset \mathbb{R}^d$ to a separable Hilbert space $\mathcal{H}$, $L \colon \mathcal{H} \to \mathbb{R}_+$ is a smooth and convex loss functional, $V \colon \Theta \to \mathbb{R}$ is the optional regularization term and $\mathcal{M}_+(\Theta)$ is the set of positive measures over the parameter set.

We observe straight away that the optimization problem (5.1) is convex as long as the loss functional L is convex. However, the optimization is defined over a set of measures, an infinite dimensional space, making the conventional convex optimization approach not applicable. Nevertheless, an interesting behavior emerges as one parametrizes the measure μ as a sum of finitely many particles – *an atomic measure* – and defines a gradient field that sets the dynamics of the particles called the *particle gradient flow*. It can be shown that in the many-particle limit, the particle gradient flow of the objective functional F provably converges to a global minimizer of (5.1).

The interplay between the many-particle limit and its asymptotic regime is a widely studied principle in deep learning theory to explain the success of SGD in terms of convergence [31, 114, 141, 133], regularization properties [32] and also the convergence of SGD in the case of continuous limit of residual networks [105].

Many papers in the literature make use of the 2-homogeneity property which restricts the analysis to the case of a neural network with a single hidden layer. We show that several of the results for single hidden layer networks also apply in the case of a wider set of functions Φ which we call *path-homogeneous*. They are defined by the existence of a vector $\alpha \in \mathbb{R}^d$ whose entries are greater or equal to one and a real number k greater or equal to the biggest entry in α such that the following holds

$$\Phi\left(\lambda^\alpha \odot \theta\right) = \lambda^k \Phi(\theta), \;\; \forall \theta \in \Theta\,, \forall \lambda > 0,$$

where $\odot$ is the element-wise Hadamard product and λ^α denotes a vector whose i^{th} entry is λ^{α_i}.

The above notion of path-homogeneity mathematically captures the importance of parameter scaling in neural networks depending on the depth. The same scaling property plays a vital role in several of the successful optimization heuristics such as batch normalization [79] or Path-SGD [117].

5.1.1 Outlook and contributions

- In Section 5.2, we formally introduce the notion of *path-homogeneity* and show that multi-layer ReLU networks fall into this category of functions.

- In Section 5.3, we prove a global convergence result of the many-particle limit for path-homogeneous models. We show that similar techniques can be used to prove a global convergence result for continuous-depth residual networks studied in [105].

- In Section 5.4, we introduce a stability-based approach to compute an *a priori* upper bound on the generalization error for the Wasserstein gradient flow for 2–homogeneous models.

- In Section 5.5, we perform numerical experiments on multi-layer ReLU networks showing that the asymptotic regime occurs at a small number of particles that increases with the depth of the network. We observe that increasing the number of particles for a deep convolutional neural network applied to CIFAR-10 classification empirically improves the test accuracy even if the training loss remains the same.

5.1.2 Related work

Mean-field limit of two-layer neural networks. A number of works use (5.1) and choose μ to be an atomic measure with m particles to model a two-layer neural network with m hidden states. Using this framework [114] and [141] proved that the dynamics of the gradient descent training of a two-layer neural network converges to a well-defined PDE as m tends to infinity. Furthermore, the work of [133] proves the long-term convergence of gradient descent to the true model in the large data limit at a rate m^{-1} as $m \to \infty$. Finally, [31] showed convergence to a global minimizer in the small data regime using 2-homogeneity of the network and a uniform initialization. Our work extends the results of [31] to the case where each particle represents practical architectures of neural networks that have **arbitrary depth**, that can include **biases**, **batch normalization** and **pooling layers**.

Ensemble averaging. Ensemble averaging is a simple technique to improve the test accuracy of a model that is often used, for example in the winning solutions of data science competitions. The idea is that by combining outputs of multiple models, it is possible to reduce the variance of the prediction by averaging out inaccuracies, thereby improving generalization. Solving (5.1) with μ restricted to an atomic measure with m particles can be interpreted as ensemble averaging over m models defined by Φ. Note however, that applying the particle gradient flow on an atomic measure does not correspond to training each of the models independently as is usually done in ensemble training. It takes into account correlation between each of the models' predictions and is close in spirit to the idea of negative correlation learning of ensembles [102]. Our experiments show that the particle gradient flow is a suitable method for training average ensembles of deep neural networks. We empirically observe that as we **increase the number of particles**, the average ensemble continues to **decrease the test loss** even if its loss on the train dataset plateaus.

Residual network as an ensemble of shallow models. Adding skip connections between layers in neural networks introduced models that are of several orders of magnitude deeper than was previously feasible. By allowing the inputs to bypass layers, residual networks challenged the conventional way of thinking about learning models as a strict pipeline sequence. It is widely believed that the effective depth of residual networks is much lower than the number of residual layers and it has been observed that residual networks behave like an ensemble of shallow networks [150].

The work of [105] introduces the continuum limit of residual networks modeled by a differential equation and proves a global convergence of gradient descent as the number of particles goes to infinity. However, the proof only accommodates for residual blocks with a single layer which does not correspond to practice. Using the notion of path-homogeneity we are able to extend this result to allow for **multi-layer residual blocks** with **biases** and **batch-normalization**.

Optimization with scale invariant parameters. Training a deep neural network is a challenging optimization problem and various heuristics emerged to speed it up. For example, batch normalization (BN) is a widely used approach that rescales weights between layers [79]. In a similar vein, the work of [117] introduces Path-SGD, an optimization method that preserves the scaling property of ReLU networks while keeping the input-output map unchanged, leading to better empirical performance compared to SGD or AdaGrad. Our notion of path-homogeneity mathematically factors the scaling invariance property of neural networks into the convergence analysis of Wasserstein gradient flows.

Generalization for overparametrized two-layer neural networks. When the number of parameters far exceeds the number of samples, the standard Vapnik–Chervonenkis bounds become vacuous. In this regime, norm-based have been introduced to study the generalization properties of 2–layer neural networks through the lens of Rademacher complexity, for example [17, 156]. However, empirical studies [85] show little correlation between norms/margins and generalization. Uniform stability of the optimization procedure has also been introduced [24, 69], at the expense of ignoring the precise network architecture. This can lead to vacuous bounds under input or label noise [162]. More precise results [9, Theorem 5.1] proved that the excess generalization error is given by a function of Gram matrix H^∞ from a kernel associated with the ReLU activation. Also, [29] prove that if the dataset is generated by a known measure μ_{true}, then the generalization bound is given by the χ^2 distance between μ_{true} and the initialization μ_0. However, the latter two bounds rely on strong assumptions on the optimization procedure, namely that the weights stay in the *lazy* regime, that is, close to initialization.

5.1.3 Notation

We denote $\mathcal{M}(\Theta)$ to be the set of measures over the parameter set Θ (or $\mathcal{M}_+$ for the set of non-negative measures). For a vector $x \in \mathbb{R}^d$, define $\operatorname{diag}(x) \in \mathbb{R}^{d\times d}$ with $\operatorname{diag}(x)_{ii} = x_i$ and $\operatorname{diag}(x)_{ij} = 0$ for $i \neq j$. We denote the Mahalanobis distance for a positive semi-definite matrix $A \in \mathbb{R}^{d\times d}$ as $\|x\|_A := \sqrt{x^T Ax}$. The space of continuous functions from $\mathcal{X}$ to $\mathcal{Y}$ is denoted as $C(\mathcal{X}, \mathcal{Y})$ and $\mathcal{P}_2(\Theta)$ is the space of probability distributions over Θ with finite second moment and W_p is the p-Wasserstein distance.

5.2 Path-homogeneity

5.2.1 Definitions

Definition 5.1 (Path-homogeneous function)**.** *Let U be a real vector space and $f : \mathbb{R}^d \to U$. We say that f is* path-homogeneous *if there exists $\alpha \in \mathbb{R}^d_{\geq 1}$ and $k \in \mathbb{R}$ such that $k \geq \max_i \alpha_i$ and*

$$f(\lambda^\alpha \odot x) = \lambda^k f(x), \;\; \forall x \in \mathbb{R}^d \; \forall \lambda > 0.$$

To specify which α and k fulfill the above definition, we say that f is $(\alpha, k)-$homogeneous.

5.2.2 Path-homogeneous function

Let f be a $(\alpha, k)-$homogeneous function. Define for a fixed $x \in \mathbb{R}^d \backslash \{0\}$ the path $p_x \colon \lambda \in \mathbb{R}_{>0} \mapsto \lambda^\alpha \odot x \in \mathbb{R}^d$. We then have $f(p_x(\lambda)) = f(\lambda^\alpha \odot x) = \lambda^k f(x)$. Let $A := \operatorname{diag}(\alpha)$. Define now the ellipse

$$\mathcal{E} := \left\{x \in \mathbb{R}^d : \|x\|_A = 1\right\}. \tag{5.2}$$

The projection $\pi_{\mathcal{E}} \colon \mathbb{R}^d \backslash \{0\} \to \mathcal{E}$ along the curves $(p_x)_{x\in\mathcal{E}}$ is well-defined in the sense that for any $x \in \mathbb{R}^d \backslash \{0\}$, there is a unique $\xi \in \mathcal{E}$ such that $x \in p_\xi(\mathbb{R}_{>0})$. The reason behind choosing the ellipse $\mathcal{E}$ as a reference set is motivated by the fact that the tangent vector field of the paths p_ξ with $\xi \in \mathcal{E}$ at $\lambda = 1$ is perpendicular to $\mathcal{E}$. Thus, we can decompose nicely the gradient of an $(\alpha, k)-$homogeneous function using its normal and tangential component, see Remark C.5 in Supplemental material.

Define finally the projection operator $h \colon \mathcal{M}(\mathbb{R}^d) \to \mathcal{M}(\mathcal{E})$ such that for $\mu \in \mathcal{M}(\mathbb{R}^d)$, for every continuous and bounded function $\psi \colon \mathcal{E} \to \mathbb{R}$,

$$\int_{\mathcal{E}} \psi(\xi) \mathrm{d}h(\mu)(\xi) = \int_{\mathbb{R}^d \backslash \{0\}} \lambda(x)^k \psi(\pi_{\mathcal{E}}(x)) \,\mathrm{d}\mu(x). \tag{5.3}$$

Remark 5.2. *Note that h preserves $(\alpha, k)-$homogeneous functions, i.e. if $f\colon \mathbb{R}^d \to \mathbb{R}$ is $(\alpha, k)-$ homogeneous, then $\int_{\mathcal{E}} f \mathrm{d}h(\mu) = \int_{\mathbb{R}^d} f \mathrm{d}\mu$. It means that h properly scales down the measure on $\mathcal{E}$ for $(\alpha, k)-$homogeneous functions.*

Definition (5.1) is motivated by the fact that multi-layer neural networks with ReLU activations are path-homogeneous, as first noticed in [118].

5.2.3 Multi-layer ReLU networks

Let $\Phi(\theta) \in L^2(\mathcal{X}, \mathcal{Y})$ be a neural network from the input space $\mathcal{X}$ to the output space $\mathcal{Y}$ with parameters $\theta \in \Theta$, $K \in \mathbb{N}$ layers and widths of the hidden layers $(d_0, \ldots, d_K)$, where d_0 is the input dimension and d_K is the output dimension of the network. The neural network is of the form $\Phi(\theta)(x) = h_{\theta_K} \circ \cdots \circ h_{\theta_1}(x)$, where $\theta = (\theta_1, \ldots, \theta_K)$, and $\theta_k = (W_k, b_k)$ are the parameters of the k^{th} layer, $h_{\theta_k}(z) = \sigma(W_k z + b_k)$ is the single layer mapping and $\sigma = \max(\cdot, 0)$ is the ReLU activation function that applies element-wise.

We can readily see that Φ is path-homogeneous. Indeed, for $i = 1, \ldots, d$, if we let $\alpha_i = 1$ when θ_i corresponds to an entry of any matrix W_k, and $\alpha_i = k$ when θ_i corresponds to an entry of the bias b_k, then we have $\Phi(\lambda^\alpha \odot \theta) = \lambda^K \Phi(\theta)$. As a result a K-layer neural network Φ is (α, K)-homogeneous.

Now, the problem statement (5.1) indicates that the model of interest is not $\Phi(\theta)$ where the variable to optimize is $\theta \in \Theta$, but $\int_\Theta \Phi(\theta)\mathrm{d}\mu(\theta)$ where the variable to optimize is $\mu \in \mathcal{M}_+(\Theta)$. That means, if we take for example $\mu_m = m^{-1}\sum_{i=1}^m \delta_{\theta_i}$, then

$$\overline{\Phi}(\mu_m) \coloneqq \int_\Theta \Phi(\theta)\mathrm{d}\mu_m(\theta) = \frac{1}{m}\sum_{i=1}^m \Phi(\theta_i).$$

When Φ is a fully-connected multi-layer neural network of depth K with hidden layer widths $(d_0, \ldots, d_K)$, $\overline{\Phi}(\mu_m)$ is a multi-layer neural network of depth $K+2$ with hidden layer widths $(d_0, md_0, \ldots, md_K, d_K)$ with the following properties.

(i) The first layer is deterministic and non-trainable: it is duplicating the input $x \in \mathbb{R}^{d_0}$ into m copies.

(ii) The neuron n_1 at layer k is connected to neuron n_2 at layer $k+1$ if and only if $\lfloor n_1/d_k \rfloor = \lfloor n_2/d_{k+1} \rfloor$.

(iii) The last layer is deterministic and non-trainable: it is averaging over the outputs $\Phi(\theta_i)(x) \in \mathbb{R}^{d_K}$ for $i = 1, \ldots, m$.

The network structure of $\overline{\Phi}(\mu_m)$ is drawn in Figure 5.1. We see that the neural connections are local, and are not scaling with m. These types of networks fall under the umbrella of *locally-connected* neural networks in the literature. Convolutional neural networks are an example of such networks, with the additional constraint that weights are shared across channels.

Remark 5.3. *Observe that σ is not differentiable at 0, so Φ is not differentiable everywhere, invalidating Assumption 5.6. However, this is only a technical issue which can be circumvented by using the differentiable parametrization as was done in [31, Section 4.2] consisting of duplicating the network at each layer with its opposite sign thus ensuring differentiability around 0.*

Pooling and Batch Normalization Neural networks with pooling layers and batch normalization layers are also path-homogeneous. Indeed, a max-pooling or average-pooling layer, does not alter the $(\alpha, k)-$homogeneity of the network. On the other hand, a batch normalization [79] layer preserves the path-homogeneity of the network, but does affect the values of α and k. If $\Phi(\theta)(x)$ is a $(\alpha, k)-$homogeneous network and the input samples are $(x_b)_{b=1}^B$, then applying batch normalization (BN) to Φ yields

$$\mathrm{BN}_{\beta,\gamma} \circ \Phi(\theta)(x) = \beta + \gamma \cdot \frac{\Phi(\theta)(x) - \mathrm{mean}\left(\Phi(\theta)(x_b)\right)_{b=1}^B}{\mathrm{sd}\left(\Phi(\theta)(x_b)\right)_{b=1}^B}. \tag{5.4}$$

Thus $\theta \mapsto \mathrm{BN}_{\beta,\gamma} \circ \Phi(\theta)(x)$ becomes (α', k')-homogeneous, where $\alpha' = (\alpha, 1, 1)$ and $k' = 1$ which results into a faster observed convergence of the many-particle limit. This is in line with the empirical evidence showing that factoring in the scale of the weights in a deep ReLU network helps to overcome unbalanced initialization and improves generalization [117, 118].

5.3 Global convergence of Wasserstein gradient flow for path-homogeneous models

This section introduces the mathematical framework of particle and Wasserstein gradient flows that underpins our theoretical results. It then establishes the main convergence result of the Wasserstein gradient flow (5.8) to a global minimum of (5.1) as $t \to \infty$ in the case where the model Φ and the regularizer V are path-homogeneous.

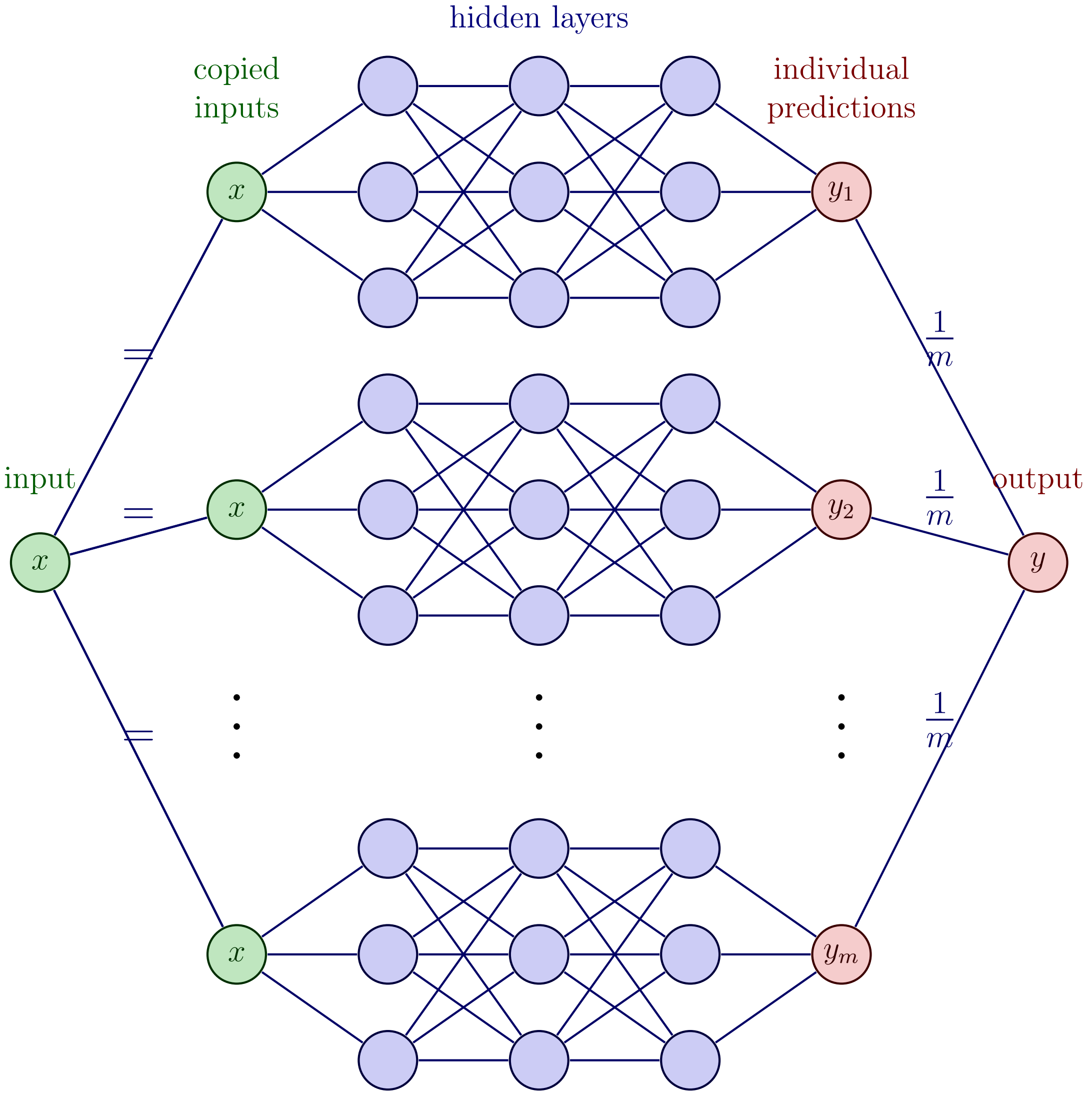

Figure 5.1: Neural network representation for the model $x \mapsto y = \overline{\Phi}(\mu_m)(x) = \int_\Theta \Phi(\theta)(x)\mathrm{d}\mu_m(\theta)$. Here, x is the input, $\Phi(\theta)$ is a neural network mapping with parameter θ, $\mu_m = \frac{1}{m}\sum_{i=1}^m \delta_{\theta_i}$ is an atomic measure over the parameter space with $\theta_1, \ldots, \theta_m \in \Theta$, $y_i = \Phi(\theta_i)(x)$ is the output of the neural network $\Phi(\theta_i)$, and $y = \frac{1}{m}\sum_{i=1}^m y_i$ is the output of the model.

5.3.1 Calculus on the space of measures

Let $\Theta \subset \mathbb{R}^d$ be the closure on an open convex set, and denote by $\mathcal{P}_p(\Theta)$ the space of probability measures on Θ that have finite p-moment, for $p \geq 1$. Let $\mu, \nu \in \mathcal{P}_p(\Theta)$. We call $\pi \in \mathcal{P}_p(\Theta \times \Theta)$ a *coupling* of μ and ν if for any Borel $B \subset \Theta$, we have $\pi(B \times \Theta) = \mu(B)$ and $\pi(\Theta \times B) = \nu(B)$, i.e. the first marginal of π is μ and the second marginal is ν. We denote by $\Gamma(\mu, \nu)$ the set of all couplings of μ and ν. We define the Wasserstein distance between μ and ν as

$$W_p(\mu, \nu) := \inf \left\{ \left(\int_{\Theta \times \Theta} \|u_1 - u_2\|^p \, \mathrm{d}\pi(u_1, u_2) \right)^{1/p} : \pi \in \Gamma(\mu, \nu) \right\}.$$

We denote the set of *optimal couplings* as $\Gamma_o(\mu, \nu)$, i.e $\pi_o \in \Gamma_o(\mu, \nu)$ if and only if $W_p^p(\mu, \nu) = \int_{\Theta \times \Theta} \|u_1 - u_2\|^p \, \mathrm{d}\pi_o(u_1, u_2)$. It turns out that we have a dual formulation for the W_1 distance given by the Kantorovitch-Rubinstein theorem [151, Theorem 1.14].

$$W_1(\mu, \nu) = \sup \left\{ \int_{\Theta} \varphi \, \mathrm{d}(\mu - \nu) \colon \phi \in L^1(\mathrm{d}\,|\mu - \nu|), \ \|\varphi\|_{\mathrm{Lip}} \leq 1 \right\}$$

where $\|\varphi\|_{\mathrm{Lip}} = \sup_{u \neq u'} \frac{|\varphi(u) - \varphi(u')|}{\|u - u'\|}$ is the Lipschitz constant of φ.

We say that a functional $F \colon \mathcal{M}_+(\Theta) \to \mathbb{R}$ is *continuously differentiable* if there exists a bounded continuous function $F' \colon \mathcal{M}_+(\Theta) \times \Theta \to \mathbb{R}$ such that for all $\mu, \nu \in \mathcal{M}_+(\Theta)$,

$$F(\mu) - F(\nu) = \int_0^1 \mathrm{d}\lambda \int_{\Theta} F'((1 - \lambda)\nu + \lambda\mu, \, u)(\mu(\mathrm{d}u) - \nu(\mathrm{d}u)). \tag{5.5}$$

We call F' the *linear functional derivative* of F. Note that (5.5) defines F' up to a constant shift, so we also impose the normalizing condition $\int_\Theta F'(\mu, u) \mathrm{d}\mu(u) = 0$ so that at most one linear functional derivative can exist. It is also denoted $\frac{\delta F}{\delta \mu}$ in the literature. Further, we call $\nabla_u F'(\mu, u)$ the *intrinsic* derivative of F.

5.3.2 Particle and Wasserstein gradient flow

If Φ is a path-homogeneous function, the minimum over the set of positive measures is equal to the minimum over the set of probability measures, $\min_{\mu \in \mathcal{M}_+(\Theta)} F(\mu) = \min_{\mu \in \mathcal{P}(\Theta)} F(\mu)$, see Lemma C.1 for a proof. Therefore it suffices to work in the space of probability distributions.

We parametrize the measure μ in (5.1) as a mixture of m particles $\mu_m = \frac{1}{m} \sum_{i=1}^m \delta_{\boldsymbol{u}_i}$ at positions $\boldsymbol{u} = \{\boldsymbol{u}_i\}_{i=1}^m \in \Theta^m$ referred to as an *atomic measure*. The objective can be expressed as

$$F_m(\boldsymbol{u}) := F\left(\frac{1}{m}\sum_{i=1}^{m}\delta_{\boldsymbol{u}_i}\right) = L\left(\frac{1}{m}\sum_{i=1}^{m}\Phi(\boldsymbol{u}_i)\right) + \frac{1}{m}\sum_{i=1}^{m}V(\boldsymbol{u}_i). \tag{5.6}$$

We now define *the particle gradient flow* for the atomic measure consisting of finitely many particles based on the derivative of (5.6).

Definition 5.4 (Particle gradient flow). *A* particle gradient flow *for the functional F_m is an absolutely continuous path $\boldsymbol{u} : \mathbb{R}^+ \to \Theta^m$ such that $\boldsymbol{u}'(t) = -m\nabla F_m(\boldsymbol{u}(t))$ for almost every $t \geq 0$.*

We have an explicit formulation of the velocity of each particle: $\boldsymbol{u}_i'(t) = v_t(\boldsymbol{u}_i(t))$, where v_t is the velocity field at time t defined as minus the intrinsic derivative of F evaluated at the empirical measure $\mu_{m,t}$:

$$v_t(u) := -\nabla F'(\mu_{m,t})(u) = -\left[\left\langle \mathrm{d}L\left(\int_\Theta \Phi \mathrm{d}\mu_{m,t}\right), \partial_j\Phi(u)\right\rangle\right]_{j=1}^{d} - \nabla V(u). \tag{5.7}$$

For a precise statement of a more general case where V can be non-smooth, see [31, Proposition 2.3]. Note that the explicit expression defining the velocity of each particle in Equation (5.7) gives us an insight at how we can extend the particle gradient flow to an arbitrary measure [31].

We now have the necessary tools to extend the definition of the classical gradient flow from Definition 5.4 to the set of measures.

Definition 5.5 (Wasserstein gradient flow). *A* Wasserstein gradient flow *for the functional F on $[0, T)$ is an absolutely continuous path $(\mu_t)_{t\in[0,T)} \subset \mathcal{P}_2(\Theta)$ which is a distributionally weak solution to the PDE*

$$\partial_t\mu_t + \operatorname{div}(v_t\mu_t) = 0, \quad \textit{where } v_t(u) = -\nabla F'(\mu_t)(u). \tag{5.8}$$

That means,

$$\int_0^T\int_\Theta\left(\frac{\partial\varphi}{\partial t}(t,u) + v_t(u)^\top\nabla_u\varphi(t,u)\right)\mathrm{d}\mu_t(u)\mathrm{d}t = 0, \ \ \forall\varphi \in C_c^\infty((0,T)\times\Theta). \tag{5.9}$$

One can show that under some mild regularity conditions, v_t belongs to the tangent vector space of μ_t if and only if the continuity equation $\partial_t\mu_t + \operatorname{div}(v_t\mu_t) = 0$ is satisfied, see [5, Proposition 8.4.5].

To establish the existence of the Wasserstein gradient flow for the functional F defined in (5.1), we make the following assumptions.

Assumption 5.6. *Denote* $\widetilde{\Phi} = \Phi\big|_{\mathcal{E}}$, $\widetilde{V} = V\big|_{\mathcal{E}}$, *and*

$$\mathcal{H}_{\mathcal{E}} = \left\{ \int_{\mathcal{E}} \widetilde{\Phi} \mathrm{d}\rho \colon \rho \in \mathcal{P}_2(\mathcal{E}) \right\}.$$

We have:

(i) $L \colon \mathcal{H} \to \mathbb{R}_+$ *is convex, differentiable and its Frechet derivative* $\mathrm{d}L$ *is bounded on* $\mathcal{H}_{\mathcal{E}}$*, i,e* $\|\mathrm{d}L\|_{\infty,\mathcal{E}} \coloneqq \sup_{h \in \mathcal{H}_{\mathcal{E}}} \|\mathrm{d}L(h)\|_{\mathcal{H}} < \infty$. *Also,* $\mathrm{d}L$ *is* $\mathrm{Lip}_{\mathrm{d}L,\mathcal{E}}$*–Lipschitz on* $\mathcal{H}_{\mathcal{E}}$ *and bounded on sublevel sets, i.e.* $\sup \{\|\mathrm{d}L(h)\|_{\mathcal{H}} : L(h) < c\} < \infty$ *for each* $c > 0$.

(ii) $\Phi \colon \Theta \to \mathcal{H}$ *is Frechet differentiable, and its Frechet derivative* $\mathrm{d}\Phi$ *is bounded on* $\mathcal{E}$*, that is,* $\|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} \coloneqq \sup_{\xi \in \mathcal{E}} \left\| \mathrm{D}_{\xi} \widetilde{\Phi} \right\| < \infty$*, and* $\mathrm{D}.\widetilde{\Phi}$ *is* $\mathrm{Lip}_{\mathrm{d}\Phi,\mathcal{E}}$*–Lipschitz on* $\mathcal{E}$.

(iii) V *is differentiable, and its gradient is bounded by* $\|\nabla V\|_{\infty,\mathcal{E}}$ *on* $\mathcal{E}$*, and is* $\mathrm{Lip}_{\nabla V,\mathcal{E}}$*–Lipschitz on* $\mathcal{E}$.

We prove now that under Assumption 5.6, there exists a Wasserstein gradient flow $(\mu_t)_{t\geq 0}$ for the functional F. However, $(\mu_t)_{t\geq 0}$ may not be unique. But the Wasserstein gradient flow on $\mathcal{E}$ is unique: to see that, define for $\rho \in \mathcal{M}_+(\mathcal{E})$ the functional

$$F_{\mathcal{E}}(\rho) \coloneqq L\left(\int_{\mathcal{E}} \Phi(\xi) \mathrm{d}\rho(\xi) \right) + \int_{\mathcal{E}} V(\xi) \mathrm{d}\rho(\xi). \tag{5.10}$$

Observe that as Φ and V are (α, k)–homogeneous, we have $F(\mu) = F_{\mathcal{E}}(h(\mu))$. That means, optimizing F is equivalent to optimizing $F_{\mathcal{E}}$. Now, there exist a *unique* gradient flow $(\rho_t)_{t\geq 0} \subset \Theta_2(\mathcal{E})$ for the functional $F_{\mathcal{E}}$.

Proposition 5.7. *Under Assumption 5.6, assume that there exists* $r_0 > 0$ *such that* $\mathrm{supp}(\mu_0) \subset r_0\mathcal{E}$. *Then there exists a Wasserstein gradient flow* $(\mu_t)_{t\geq 0}$ *starting from* μ_0 *for the functional* F. *Moreover, there exists a unique gradient flow* $(\rho_t)_{t\geq 0}$ *on* $\mathcal{P}_2(\mathcal{E})$ *such that* $\partial_t \rho_t + \mathrm{div}(v_t^{\mathcal{E}} \rho_t) = 0$*, where* $v_t^{\mathcal{E}}(\xi) = -\mathrm{D}_{\xi} F'_{\mathcal{E}}(\rho_t)$.

Here, D_{ξ} is the differential at point ξ on the manifold $\mathcal{E}$. Uniqueness of the gradient flow on $\mathcal{P}_2(\Theta)$ is not guaranteed, but uniqueness on $\mathcal{P}_2(\mathcal{E})$ is.

Proof. For the existence of a Wasserstein gradient flow, observe that F is a proper, coercive, and differentiable functional, hence [5, Corollary 11.1.8] applies here.
For the uniqueness of the gradient flow for $F_{\mathcal{E}}$, it suffices to show that $F_{\mathcal{E}}$ is $\lambda_{\mathcal{E}}$–semiconvex along generalized geodesics, for some $\lambda_{\mathcal{E}} \in \mathbb{R}$. Uniqueness of the gradient flow and an associated system of evolution variational inequalities follows from [5,

Theorem 11.1.4].

Let $\tau \in \mathcal{P}(\mathcal{E}\times\mathcal{E})$ a transport plan such that both marginals have finite second moments, and denote the cost associated to it by

$$C_p(\tau) := \left(\int_{\mathcal{E}^2} d_{\mathcal{E}}(x,y)^p \mathrm{d}\tau(x,y)\right)^{1/p}$$

for $p \geq 1$, where $d_{\mathcal{E}}(x,y)$ is the geodesic distance between x and y. Now, $\mathcal{E}$ is a compact Riemannian manifold, so there exists a geodesic $\gamma_{\cdot} \equiv \gamma_{\cdot}(x,y)\colon [0,1] \to \mathcal{E}$ such that $\gamma_0(x,y) = x$ and $\gamma_1(x,y) = y$. Let $\rho_s^\tau := (\gamma_s)_\# \tau$, and define $g(s) = F_{\mathcal{E}}(\rho_s^\tau)$ for $s \in [0,1]$. Since $\mathrm{d}L$, $\mathrm{d}\Phi$ and ∇V are Lipschitz on $\mathcal{E}$, g is differentiable with

$$\dot{g}(s) = \frac{\mathrm{d}}{\mathrm{d}s} F_{\mathcal{E}}(\rho_s^{tau}) = \left\langle \mathrm{d}L\left(\int_{\mathcal{E}} \Phi \mathrm{d}\rho_s^\tau\right), \int_{\mathcal{E}^2} \mathrm{D}_{\gamma_s}\Phi(\dot{\gamma}_s)\mathrm{d}\tau\right\rangle + \int_{\mathcal{E}^2} \mathrm{D}_{\gamma_s} V(\dot{\gamma}_s)\mathrm{d}\tau$$

Therefore, for $0 \leq s < s' \leq 1$, we can bound

$$|\dot{g}(s') - \dot{g}(s)| \leq \text{(I)} + \text{(II)} + \text{(III)} + \text{(IV)},$$

where

$$\begin{aligned}
\text{(I)} &= \left|\left\langle \mathrm{d}L\left(\int_{\mathcal{E}} \Phi \mathrm{d}\rho_s^\tau\right), \int_{\mathcal{E}^2} \mathrm{D}_{\gamma_{s'}}\Phi\left(\dot{\gamma}_{s'} - \dot{\gamma}_s\right)\mathrm{d}\tau\right\rangle\right| \\
\text{(II)} &= \left|\left\langle \mathrm{d}L\left(\int_{\mathcal{E}} \Phi \mathrm{d}\rho_s^\tau\right), \int_{\mathcal{E}^2} \left(\mathrm{D}_{\gamma_{s'}}\Phi - \mathrm{D}_{\gamma_s}\Phi\right)(\dot{\gamma}_s)\mathrm{d}\tau\right\rangle\right| \\
\text{(III)} &= \left|\left\langle \mathrm{d}L\left(\int_{\mathcal{E}} \Phi \mathrm{d}\rho_{s'}^\tau\right) - \mathrm{d}L\left(\int_{\mathcal{E}} \Phi \mathrm{d}\rho_s^\tau\right), \int_{\mathcal{E}^2} \mathrm{D}_{\gamma_{s'}}\Phi\left(\dot{\gamma}_{s'}\right)\mathrm{d}\tau\right\rangle\right| \\
\text{(IV)} &= \left|\int_{\mathcal{E}^2} \mathrm{D}_{\gamma_{s'}} V\left(\dot{\gamma}_{s'} - \dot{\gamma}_s\right)\mathrm{d}\tau\right| + \left|\int_{\mathcal{E}^2} \left(\mathrm{D}_{\gamma_{s'}} V - \mathrm{D}_{\gamma_s} V\right)(\dot{\gamma}_s)\mathrm{d}\tau\right|
\end{aligned}$$

We need bounds on the (differences of the) speed of geodesics with respect to the geodesical distance $d_{\mathcal{E}}$ between the endpoints. As $\mathcal{E}$ is diffeomorphic to the sphere S^{d-1}, the geodesics are (globally) the shortest paths between points, and the following holds.

$$\|\dot{\gamma}_s(x,y)\| = d_{\mathcal{E}}(x,y) \quad \text{and} \quad \|\dot{\gamma}_{s'}(x,y) - \dot{\gamma}_s(x,y)\| \leq d_{\mathcal{E}}(x,y)^2 \left|s' - s\right|$$

Therefore, we can bound (I) – (IV).

$$\begin{aligned}
\text{(I)} &\leq \|\mathrm{d}L\|_{\infty,\mathcal{E}} \, \|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} \, |s'-s| \int_{\mathcal{E}^2} d_{\mathcal{E}}(x,y)^2 \mathrm{d}\tau(x,y) \\
&= \|\mathrm{d}L\|_{\infty,\mathcal{E}} \, \|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} \, C_2^2(\tau) \, |s'-s| \\
\text{(II)} &\leq \|\mathrm{d}L\|_{\infty,\mathcal{E}} \, \mathrm{Lip}_{\mathrm{d}\Phi,\mathcal{E}} \int_{\mathcal{E}^2} \|\gamma_{s'}(x,y) - \gamma_s(x,y)\| \, \|\dot{\gamma}_s\| \, \mathrm{d}\tau(x,y) \\
&\leq \|\mathrm{d}L\|_{\infty,\mathcal{E}} \, \mathrm{Lip}_{\mathrm{d}\Phi,\mathcal{E}} C_2^2(\tau) \, |s'-s| \\
\text{(III)} &\leq \mathrm{Lip}_{\mathrm{d}L,\mathcal{E}} \left\| \int_{\mathcal{E}^2} (\Phi(\gamma_{s'}) - \Phi(\gamma_s)) \, \mathrm{d}\tau \right\| \|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} \int_{\mathcal{E}^2} \|\dot{\gamma}_{s'}\| \, \mathrm{d}\tau \\
&\leq \mathrm{Lip}_{\mathrm{d}L,\mathcal{E}} \, \|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} \, C_1^2(\gamma) \, |s'-s| \\
\text{(IV)} &\leq \|\nabla V\|_{\infty,\mathcal{E}} \, C_2^2(\gamma) \, |s'-s| + \mathrm{Lip}_{\nabla V,\mathcal{E}} C_2^2(\gamma) \, |s'-s|
\end{aligned}$$

As a consequence, using $C_1(\gamma) \leq C_2(\gamma)$ by Cauchy-Schwarz, we get that $\dot{g}$ is $\lambda_{\mathcal{E}} C_2^2(\gamma)$–Lipschitz, with

$$\lambda_{\mathcal{E}} = \|\mathrm{d}L\|_{\infty,\mathcal{E}} \, \|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} + \|\mathrm{d}L\|_{\infty,\mathcal{E}} \, \mathrm{Lip}_{\mathrm{d}\Phi,\mathcal{E}} + \mathrm{Lip}_{\mathrm{d}L,\mathcal{E}} \, \|\mathrm{d}\Phi\|_{\infty,\mathcal{E}} + \|\nabla V\|_{\infty,\mathcal{E}} + \mathrm{Lip}_{\nabla V,\mathcal{E}}.$$

Hence, by definition, $F_{\mathcal{E}}$ is $\lambda_{\mathcal{E}}$–semiconvex along generalized geodesics.

□

Remark 5.8. *Under Assumption 5.6, the particle gradient flow of m particles in Definition 5.4 converges to a Wasserstein gradient flow as $m \to \infty$, see [31, Theorem 2.6.].*

We can see from Definition 5.5 that a probability measure $\mu \in \mathcal{P}_2(\Theta)$ is stationary if and only if $\nabla F'(\mu)(u) = 0$ for μ–a.e. $u \in \Theta$. But stationarity does not imply optimality, even when L is convex. Instead, one has the following characterization, taken from [121].

Lemma 5.9 (Optimality condition)**.** *The measure $\mu^* \in \mathcal{M}_+(\Theta)$ is a minimizer of F if and only if $F'(\mu^*)(u) \geq 0$ for all $u \in \Theta$ and $F'(\mu^*)(u) = 0$ for $u \in \mathrm{supp}(\mu^*)$.*

Proof. Suppose first that $\mu^* \in \mathcal{M}_+(\Theta)$ is a minimizer of $F(\mu) = L\left(\int \Phi \mathrm{d}\mu\right) + \int V \mathrm{d}\mu$. Thus, for all $\mu \in \mathcal{M}_+(\Theta)$, we have that

$$\int F'(\mu^*) \mathrm{d}(\mu - \mu^*) = \frac{\mathrm{d}}{\mathrm{d}\epsilon} F((1-\epsilon)\mu^* + \epsilon\mu)\big|_{\epsilon=0} = \lim_{\epsilon \to 0} \frac{F((1-\epsilon)\mu^* + \epsilon\mu) - F(\mu^*)}{\epsilon} \geq 0.$$

By letting $\mu = \mu^* + \delta_u$ for a fixed $u \in \Theta$, we deduce first that $F'(\mu^*)(u) \geq 0$ for all $u \in \Theta$. Next, by letting μ be the zero measure, we obtain $\int F'(\mu^*) \mathrm{d}\mu^* = 0$, so by positivity of μ^*, $F'(\mu^*)(u) = 0$ for $u \in \mathrm{supp}(\mu^*)$.

To show the converse, first observe that the two conditions imply that $\int F'(\mu^*)\mathrm{d}(\mu - \mu^*) \geq 0$. As L is convex, F is convex and we have

$$\begin{aligned} F(\mu) - F(\mu^*) &= \frac{\mathrm{d}}{\mathrm{d}\epsilon}((1-\epsilon)F(\mu^*) + \epsilon F(\mu))\,\big|_{\epsilon=0} \\ &\geq \frac{\mathrm{d}}{\mathrm{d}\epsilon}F((1-\epsilon)\mu^* + \epsilon\mu)\big|_{\epsilon=0} = \int F'(\mu^*)\mathrm{d}(\mu - \mu^*) \geq 0. \end{aligned}$$

So μ^* is a global minimizer. □

5.3.3 Convergence to the global minimum

We make the following assumptions to prove the global convergence of the Wasserstein gradient flow.

Assumption 5.10. *Φ and V are $(\alpha, k)-$homogeneous and the support of $h(\mu_0)$ is the whole of $\mathcal{E}$.*

We stress that Φ and V share the same α and k. The fact that $\mathrm{supp}(h(\mu_0)) = \mathcal{E}$ ensures that the measure allocates a non-zero mass everywhere in the ellipse at least during a short period of time to ensure that the gradient flow does not miss any important region of the parameter space. This assumption is also used by [34] to prove global convergence.

Proposition 5.11 (Wasserstein gradient flow escapes local minima)**.** *Let Assumption 5.6 and Assumption 5.10 hold true and let $\mu \in \mathcal{M}_+(\Theta)$ be a measure such that $\{\theta \in \Theta : F'(\mu)(\theta) < 0\}$ is not empty. Then there exists $\epsilon > 0$ and a subset of the parameter set $P \subset \Theta$ such that if $(\mu_t)_{t\geq 0}$ is the Wasserstein gradient flow of F satisfying $\|h(\mu) - h(\mu_{t_0})\|_{BL} < \epsilon$ for some $t_0 \geq 0$ and $\mu_{t_0}(P) > 0$, then there also exists $t_1 > t_0$ such that $\|h(\mu) - h(\mu_{t_1})\|_{BL} > \epsilon$.*

The proof of Proposition 5.11 can be found in Appendix C.2.2. We now establish that for the projected Wasserstein gradient flow that converges weakly to the measure ν, F' evaluated at ν vanishes on its support.

Proposition 5.12 (F' vanishes for the limit of the projected Wasserstein gradient flow)**.** *Let $(\mu_t)_{t\geq 0}$ be a Wasserstein gradient flow of F and let Assumption 5.6 and Assumption 5.10 hold. If $h(\mu_t)$ converges weakly to $\nu \in \mathcal{M}_+(\mathcal{E})$, then $F'(\nu)$ vanishes ν almost surely.*

Proof. As $F'(\mu_t)$ is an $(\alpha, k)-$homogeneous function, we know by Remark C.5 that the velocity field $(v_t)_{t\geq 0} = (-\nabla F'(\mu_t))_{t\geq 0}$ associated to the gradient flow $(\mu_t)_{t\geq 0}$ satisfies

$$\forall \xi \in \mathcal{E} : \quad -v_t(\xi) = \frac{k\tilde{g}_{\mu_t}(\xi)}{\|A\xi\|_2}\vec{n}_\xi + \iota(\nabla \tilde{g}_{\mu_t}(\xi)) + \nabla V(\xi), \tag{5.11}$$

where A is a diagonal matrix with entries $A_{i,i} = \alpha_i$. Hence by Lemma C.6, v_t converges uniformly to $v_\nu = -\nabla F'(\nu)$ on $\mathcal{E}$ as $t \to \infty$, so in particular

$$\int_{\mathcal{E}} F'(\mu_t)(\xi)\mathrm{d}h(\mu_t)(\xi) \to \int_{\mathcal{E}} F'_\nu(\xi)\mathrm{d}\nu(\xi). \tag{5.12}$$

We first use the conservation of energy for the Wasserstein gradient flow [5, Theorem 11.2.1.] to obtain

$$\forall t \geq 0 : \quad -\frac{\mathrm{d}}{\mathrm{d}t}F(\mu_t) = \int_\Theta |\nabla F'(\mu_t)(u)|^2 \,\mathrm{d}\mu_t(u) \tag{5.13}$$

We use Cauchy-Schwarz on the right-hand side of Equation (5.13) and Remark C.4 to deduce that

$$\begin{aligned} -\frac{\mathrm{d}}{\mathrm{d}t}F(\mu_t) &\geq \frac{\left(\int_\Theta \langle \nabla F'(\mu_t)(u), Au\rangle \mathrm{d}\mu_t(u)\right)^2}{\int_\Theta \|Au\|_2^2 \,\mathrm{d}\mu_t(u)} \qquad (5.14)\\ &= k^2\frac{\left(\int_{\mathcal{E}} F'(\mu_t)(\xi)\mathrm{d}h(\mu_t)(\xi)\right)^2}{\int_\Theta \|Au\|_2^2 \,\mathrm{d}\mu_t(u)}. \qquad (5.15)\end{aligned}$$

We find a uniform upper bound on the denominator of (5.15) as follows.

$$\int_\Theta \|Au\|_2^2 \,\mathrm{d}\mu_t(u) \leq k^2 \int_\Theta \|u\|^2 \,\mathrm{d}\mu_t(u) \leq 2k^2 \left(\int_\Theta \|u\|_2^2 \,\mathrm{d}\mu_0(u) + F(\mu_0)\right) =: C_2 < \infty.$$

The first inequality holds as $\max_i \alpha_i \leq k$ and the second one by Lemma C.7 with $\psi(u) = \|u\|_2^2$ and $C = 4$. We deduce that

$$-\frac{\mathrm{d}}{\mathrm{d}t}F(\mu_t) \geq \frac{k^2}{C_2}\left(\int_{\mathcal{E}} F'(\mu_t)(\xi)\mathrm{d}h(\mu_t)(\xi)\right)^2. \tag{5.16}$$

But F is lower bounded, so the right-hand side of Equation (5.16) must vanish as $t \to \infty$. Thus, by Equation (5.12), we get that $F'(\nu)$ vanishes ν almost surely. □

By Lemma 5.9, it remains to show that $F'(\nu)$ is non-negative everywhere. Note that instead of the separation assumption on the initialization μ_0 introduced in [31, Theorem 3.3], we enforce the full support of $h(\mu_0)$ on $\mathcal{E}$, introduced in [34, Theorem 2.2]. It is a slightly stronger assumption, but the proof simplifies considerably. Note, that this is only a technical issue and our results would hold also under a more general assumption of separability of ellipses similar to the assumption made for [31, Theorem 3.3], but we omit the proof here for simplicity.

Proposition 5.13 (F' is non-negative everywhere)**.** *Assume that $h(\mu_0)$ has full support in $\mathcal{E}$ and that $h(\mu_t)$ converges weakly to $\nu \in \mathcal{M}_+(\mathcal{E})$. Then $F'(\nu)(\xi) \geq 0$ for all $\xi \in \mathcal{E}$.*

Proof. For the sake of contradiction, assume that $F'(\nu)$ is not non-negative everywhere. Let $\epsilon > 0$ and $P = \pi_{\mathcal{E}}^{-1}(K) \subset \Theta$ given by Proposition 5.11, and let $t_0 = \sup\{t \geq 0 : \|h(\mu_t) - \nu\|_{BL} \geq \epsilon\}$, which is finite because $h(\mu_t)$ converges weakly to ν. But $h(\mu_{t_0})$ has full support as $X_{t_0} : \Theta \to \Theta$ defined in Equation (C.15) is a diffeomorphism and $\mu_{t_0} = (X_{t_0})_{\#}\mu_0$. So $h(\mu_{t_0})(K) > 0$, thus $\mu_{t_0}(P) > 0$ and Proposition 5.11 applies, leading to a contradiction. □

Theorem 5.14. *Under Assumptions 5.6 and 5.10, if $h(\mu_t)$ converges weakly, then its limit is a global minimizer of F over $\mathcal{M}_+(\Theta)$. In particular, if $(\boldsymbol{u}_m(\cdot))_{m\in\mathbb{N}}$ is a sequence of particle gradient flows initialized in the support of μ_0 such that $\mu_{m,0}$ converges weakly to μ_0, then*

$$\lim_{m,t\to+\infty} F(\mu_{m,t}) = \min_{\mu\in\mathcal{M}_+(\Theta)} F(\mu)$$

and the left-hand side limits can be exchanged.

Proof. We now show that Theorem 5.14 is a consequence of the above results. Indeed, assume that $\nu \in \mathcal{M}_+(\mathcal{E})$ is the weak limit of $h(\mu_t)_t$ as $t \to \infty$. Proposition 5.12 ensures that $F'(\nu)$ vanishes on its support, and Proposition 5.13 ensures that $F'(\nu)$ is non-negative everywhere. Thus, by Lemma 5.9, ν is a global minimizer of F. The fact that we can permute the limits $m \to \infty$ and $t \to \infty$ can be proven similarly to [31, Theorem 3.3]. □

Remark 5.15. *We stress that the result here is a qualitative one. A step towards obtaining quantitative convergence rate without injecting exogenous noise like in [77, 114] would be to study the dynamics of the projected gradient flows like in [34, Theorem 3.8]. We leave precise statements for future work.*

5.3.4 Global convergence for continuous-depth residual networks

In this section, we show that path-homogeneity can be used in the context of the mean-field limit of *continuous-depth residual networks* studied in [105]. Let $\mathcal{X} = \mathcal{Y} \subset \mathbb{R}^{d_0}$ be the input and the output space, $\mathcal{D} \subset \mathcal{X} \times \mathcal{Y}$ an input-output distribution and $\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ a loss function. Let $\Theta \subset \mathbb{R}^d$ be a set of parameters, and $\sigma\colon \Theta \times \mathcal{X} \to \mathcal{X}$ the hidden state mapping that takes the parameter of a single layer, together with its hidden state, and outputs the hidden state at the next layer. As we will work with the continuous-depth limit, we extend the parameter space to $\Theta \times [0,1]$. The second parameter $t \in [0,1]$ indicates at which (infinitesimal) layer the first parameter $\theta \in \Theta$ is acting on. Similarly to the relaxation in (5.1), we consider $\mu \in \mathcal{M}_+ (\Theta \times [0,1])$. The output of the continuous-depth residual network is then given by the solution at depth $t = 1$ of the ODE

$$\dot{Z}_\mu(x,t) = \int_\Theta \sigma(Z_\mu(x,t), \theta) \mathrm{d}\mu(\theta, t), \quad Z_\mu(x,0) = x.$$

The loss functional is given by $E(\mu) := \mathbb{E}_{(X,Y)\sim\mathcal{D}} [\ell(Z_\mu(X,1), Y)]$. Note that unlike [105], we omit the projection of the input and the output to a hidden space, as it does not influence the theoretical results. We extend [105, Theorem 3.9] about the convergence to a global minimum to include biases and batch normalization in the residual blocks using path-homogeneity, which is closer to what is done in practice.

Theorem 5.16 (Continuous-depth residual networks with path-homogeneous blocks)**.** *Let $\sigma : \Theta \times \mathcal{X} \to \mathcal{X}$ be the residual block which is (α, k)-homogeneous with respect to the parameters and $(\mu_s)_{s\geq 0}$ be the solution of the Wasserstein gradient flow of the mean-field model for continuous-depth residual networks:*

$$\frac{\partial_{(\theta,t)}\mu}{\partial s} = \mathrm{div}_{(\theta,t)} \left(\mu \cdot \nabla_{(\theta,t)} \frac{\delta L}{\delta \mu} \right). \tag{5.17}$$

Consider a stationary solution to the gradient flow μ_∞ which concentrates in one of the nested sets from Assumption 5.6 (iii) and separates the ellipses $\lambda_a \mathcal{E} \times [0,1]$ and $\lambda_b \mathcal{E} \times [0,1]$ where $\mathcal{E}$ is defined as in (5.2). Then μ_∞ is a global minimum and satisfies $F(\mu_\infty) = 0$.

Proof. This proof follows the same line of arguments as the proof in [105, Theorem 3.9.] with differences in equations (5.19), (5.20) and (5.21).

By [121] and μ_∞ being a steady state, the following must hold

$$\nabla_{(\theta,t)} \frac{\delta F}{\delta \mu}\bigg|_{\mu=\mu_\infty} = 0, \tag{5.18}$$

μ_∞-almost everywhere.

By the homogeneity property and the separation property of the support of μ_∞, we prove that $\nabla_{(\theta,t)} \frac{\delta F}{\delta \mu}|_{\mu=\mu_\infty} = 0$ almost everywhere, so also outside of the support of μ_∞. Indeed, by the separation of ellipses assumption, for any $(\theta, t) \in \mathbb{R}^{d_1 \times d_1} \times [0,1]$, there exists $\lambda > 0$ such that $(\lambda^\alpha \odot \theta, t) \in \mathrm{supp}(\mu_\infty)$. By the $(\alpha, k)-$homogeneity of σ, we have that

$$\begin{aligned}\frac{\delta F}{\delta \mu}\left(\lambda^{\alpha}\odot\theta,t\right) &= \mathbb{E}\left[\sigma\left(Z_{\mu}(x,t),\lambda^{\alpha}\odot\theta\right)\right]p_{\mu}(x,t) \qquad (5.19)\\ &= \lambda^{k}\mathbb{E}\left[\sigma\left(Z_{\mu}(x,t),\theta\right)\right]p_{\mu}(x,t) \qquad (5.20)\\ &= \lambda^{k}\frac{\delta F}{\delta \mu}\left(\theta,t\right), \qquad (5.21)\end{aligned}$$

where $p_{\mu}(x,t)$ is the adjoint process as defined in [105]. As a consequence we have that $\nabla_{(\theta,t)}\frac{\delta F}{\delta\mu}(\lambda^{\alpha}\odot\theta)=\lambda^{k}\nabla_{(\theta,t)}\frac{\delta F}{\delta\mu}(\theta,t)$.
We therefore deduce that since $\nabla_{(\theta,t)}\frac{\delta F}{\delta\mu}|_{\mu=\mu_{\infty}}=0$, $\mu_{\infty}-$a.e., we have $\nabla_{(\theta,t)}\frac{\delta F}{\delta\mu}|_{\mu=\mu_{\infty}}=0$ almost everywhere. Therefore the differential is constant: $\frac{\delta F}{\delta\mu}|_{\mu=\mu_{\infty}}=c$.
By [105, Theorem 3.2.], if $F(\mu_{\infty})>0$, there exists a distribution $\nu\in\mathcal{P}\left(\Theta\times[0,1]\right)$ such that

$$\left\langle \frac{\delta F}{\delta\mu}\bigg|_{\mu=\mu_{\infty}},\,\mu-\nu\right\rangle > 0. \qquad (5.22)$$

This is however in contradiction with

$$\left\langle \frac{\delta F}{\delta\mu}\bigg|_{\mu=\mu_{\infty}},\,\mu-\nu\right\rangle = c\left(\int_{\Theta}\mu\left(\theta,t\right)\mathrm{d}\theta\mathrm{d}t-\int_{\Theta}\nu\left(\theta,t\right)\mathrm{d}\theta\mathrm{d}t\right)=0, \qquad (5.23)$$

due to the probability measures integrating to one. Thus the stationary solution must satisfy $F(\mu_{\infty})=0$ implying that it is a global optimum. □

5.4 Generalization properties in the 2-homogeneous case

We now reformulate the minimization problem in Equation (5.1) as a supervised learning problem on $\mathcal{M}_{+}(\Theta)$, and we study the generalization properties of the solution found by the Wasserstein gradient flow. We first define the framework in which we operate.

5.4.1 Assumptions and definitions

Let $\mathcal{X}$ be the input space, $\mathcal{Y}$ the output space, and assume that the support of the data distribution $\mathcal{D}$ is bounded in $\mathcal{X}\times\mathcal{Y}$. A model is a function ϕ that takes an input $x\in\mathcal{X}$ and a parameter $\theta\in\Theta$, and gives the output $\phi(\theta,x)\in\mathcal{Y}$. The goal of a supervised learning problem is to find a (parametric) function that explain the output the best given the input, as measured by a loss function $\ell\colon\mathcal{Y}\times\mathcal{Y}\to\mathbb{R}_{+}$. In our measure-theoretic framework, the input-output function is parametrized by a measure

$\mu \in \mathcal{M}_+(\Theta)$ over the parameter space: for an input $x \in \mathcal{X}$, the prediction is given by $\int_\Theta \phi(\theta, x)\mathrm{d}\mu(\theta) \in \mathcal{Y}$. The objective function to minimize is thus

$$R(\mu) := \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\ell\left(\int_\Theta \phi(\theta, X)\mathrm{d}\mu(\theta), Y\right)\right]. \tag{5.24}$$

over $\mu \in \mathcal{M}_+(\Theta)$. We call $R(\mu)$ the *population risk* of a measure μ. However, minimizing R is intractable, as we usually only have access to i.i.d. samples $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$, where $(x_i, y_i) \sim \mathcal{D}$. Therefore, we can only evaluate the *empirical risk* of a measure μ on the dataset $\mathcal{S}_n$, as defined by

$$R_{\mathrm{emp}}(\mu, \mathcal{S}_n) := \frac{1}{n}\sum_{i=1}^n \ell\left(\int_\Theta \phi(\theta, x_i)\mathrm{d}\mu(\theta), y_i\right).$$

Optimizing $R_{\mathrm{emp}}(\mu, \mathcal{S}_n)$ directly is risky, as we might overfit to the sample set $\mathcal{S}_n$, and the fitted model might perform poorly on the full distribution $\mathcal{D}$. To circumvent this, we add a regularization term $V\colon \Theta \to \mathbb{R}_+$, with regularization coefficient $\rho_n > 0$. Our objective function can thus be written as

$$\min_{\mu\in\mathcal{M}_+(\Theta)} F_n(\mu, \mathcal{S}_n) := R_{\mathrm{emp}}(\mu, \mathcal{S}_n) + \rho_n \int_\Theta V(\theta)\mathrm{d}\mu(\theta) \tag{5.25}$$

Note that the regularization coefficient ρ_n depends on the sample size. This is because all other things being equal, the more samples we have, the closer our empirical data distribution is to $\mathcal{D}$, so the less we need to regularize.

We can rewrite (5.25) in the framework of (5.1): let $\Phi\colon \Theta \to C(\mathcal{X}, \mathcal{Y})$ defined by $\Phi(\theta)(x) := \phi(\theta, x)$, and $L\colon C(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}_+$ defined by $L(f) := \frac{1}{n}\sum_{i=1}^n \ell(f(x_i), y_i)$. We thus have

$$R_{\mathrm{emp}}(\mu, \mathcal{S}_n) = L\left(\int_\Theta \Phi(\theta)\mathrm{d}\mu(\theta)\right).$$

A important task in supervised learning is to understand how far away $R(\mu)$ is to $R_{\mathrm{emp}}(\mu, \mathcal{S}_n)$, for a given μ. One way to quantify this is to find an upper bound to

$$\mathbb{P}_{\mathcal{S}_n\sim\mathcal{D}_n}\Big(R(\mu) - R_{\mathrm{emp}}(\mu, \mathcal{S}_n) > \epsilon\Big), \tag{5.26}$$

where $\mathcal{D}_n$ is the uniform distribution over i.i.d. datasets of size n sampled from $\mathcal{D}$, and $\epsilon > 0$. We assume 2–homogeneity of the model and its regularizer, and Lipschitz continuity on the d–dimensional sphere $\mathcal{E} = S^{d-1}$, which is usually satisfied in practice.

Assumption 5.17. *Denote $\widetilde{\phi} = \phi\big|_{S^{d-1}\times\mathcal{X}}$ and $\widetilde{V} = V\big|_{S^{d-1}}$. Under the setup described above, we have for each $x \in \mathcal{X}$:*

(i) $\phi(\,\cdot\,, x)$ and V are 2–homogeneous,

(ii) $\phi(\,\cdot\,, x)$ *is Frechet differentiable, and the derivative* $\mathrm{D}.\widetilde{\phi}(\,\cdot\,, x)$ *is uniformly bounded by* $\|\mathrm{D}\phi\|_{S^{d-1},\infty} < \infty$*, and is* $\mathrm{Lip}_{\mathrm{D}\phi,\, S^{d-1},\infty}$*–Lipschitz continuous,*

(iii) $\ell\colon \mathcal{Y}\times\mathcal{Y} \to \mathbb{R}_+$ *is convex in its first variable, with* $\ell(y,y) = 0$*,* $\forall y \in \mathcal{Y}$*,* $\|\ell\|_{C^2} := \|\partial^2\ell/\partial y_1^2\|_\infty < \infty$*, and there exists* $C_\ell > 0$ *such that* $\|\partial\ell/\partial y_1\|^2 \le C_\ell^2\,\ell$*,*

(iv) V *is differentiable, and the derivative* $\mathrm{D}.\widetilde{V}$ *is uniformly bounded by* $\|\mathrm{D}V\|_{S^{d-1}}$*, and is* $\mathrm{Lip}_{\mathrm{D}V,\, S^{d-1}}$*–Lipschitz continuous.*

Assumption *(iii)* is satisfied for the quadratic loss $\ell(y_1,\, y_2) = \|y_1 - y_2\|^2$ for $C_\ell = 2$.

5.4.2 Preliminary analysis

In the following, we study the *generalization gap* (5.26) where μ is an approximation of the stationary measure of the Wasserstein gradient flow $(\mu_t)_{t\ge0}$. More specifically, for a training time $T > 0$, let $\mathcal{A}^T_{\mathrm{wgf}}\colon \mathrm{supp}(\mathcal{D})^n \to \mathcal{P}_2(\Theta)$ denote the algorithm that takes a training set $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n \sim \mathcal{D}_n$ (and an initialization $\mu_0 \in \mathcal{P}_2(\Theta)$), and outputs the measure $\mathcal{A}^T_{\mathrm{wgf}}(\mathcal{S}_n) := \mu_T$. Recall that $(\mu_t)_{t\ge0}$ is a Wasserstein gradient flow defined in (5.8) for the objective functional $F_n(\mu, \mathcal{S}_n)$ defined in (5.25).

Our approach to bound (5.26) relates to the stability approach of generalization introduced in [24], see also [88, 52]. For a dataset $\mathcal{S}_n \in \mathrm{supp}(\mathcal{D})^n$ and $i \in \{1,\dots,n\}$, we denote the *removal* of sample (x_i, y_i) from $\mathcal{S}_n$ by

$$\mathcal{S}_n^{\setminus i} := \{(x_{i'}, y_{i'})\colon i' \neq i\} \tag{5.27}$$

Definition 5.18 ([24], Definition 6)**.** *An algorithm* $\mathcal{A}$ *has uniform stability* β *with respect to the loss function* ℓ *if for each* $\mathcal{S}_n \in \mathrm{supp}(\mathcal{D})^n$*,* $i \in \{1,\dots,n\}$*, and* $(x,y) \in \mathrm{supp}(\mathcal{D})$*, we have*

$$\left|\ell\left(\int_\Theta \phi(\theta,x)\mathrm{d}\mathcal{A}(\mathcal{S}_n)(\theta),\, y\right) - \ell\left(\int_\Theta \phi(\theta,x)\mathrm{d}\mathcal{A}(\mathcal{S}_n^{\setminus i})(\theta),\, y\right)\right| \le \frac{\beta}{n}.$$

Uniform stability guarantees tight exponenential generalization bounds using concentration inequalities [24, 41].

Proposition 5.19 ([24], Theorem 12)**.** *Let* $\mathcal{A}$ *be an algorithm with uniform stability* β *with respect to* ℓ*, which satisfies*

$$\ell\left(\int_\Theta \phi(\theta,x)\mathrm{d}\mathcal{A}(\mathcal{S})(\theta),\, y\right) \in [0, B]\,, \quad \forall S \subset \mathrm{supp}(\mathcal{D}),\ \forall (x,y) \in \mathrm{supp}(\mathcal{D}). \tag{5.28}$$

Then, for $\delta \in (0,1)$*, with probability* $1-\delta$ *over the choice of the dataset* $\mathcal{S}_n$*, we have*

$$R\left(\mathcal{A}(\mathcal{S}_n)\right) \le R_{\mathrm{emp}}(\mathcal{A}(\mathcal{S}_n), \mathcal{S}_n) + \frac{2\beta}{n} + (4\beta + B)\sqrt{\frac{\log 1/\delta}{2n}}.$$

To prove uniform stability of the Wasserstein gradient flow algorithm $\mathcal{A}^T_{\mathrm{wgf}}$, we first study the dynamics of $h(\mu_t)$. We prove that $t \mapsto h(\mu_t)$ solves an advection-reaction equation [57]. A similar version of the result below was first established in [34, Proposition 2.1]; we provide a more detailed proof.

Lemma 5.20. *Let F defined in (5.1) and $F_{S^{d-1}}$ defined in (5.10), where Φ and V are 2–homogeneous. Let $(\mu_t)_{t\geq 0} \subset \mathcal{P}_2(\Theta)$ be a gradient flow for the functional F, and define $M_0 \coloneqq \int_\Theta \|u\|^2 \, \mathrm{d}\mu_0(u) < \infty$. Then $\rho_t = M_0^{-1} h(\mu_t) \in \mathcal{P}_2(S^{d-1})$ solves the following (distributional) PDE.*

$$\partial_t \rho_t + \operatorname{div}(v_t^S \rho_t) = 2F'_{S^{d-1}}(M_0\rho_t)\rho_t, \tag{5.29}$$

where $v_t^S(\xi) = -\mathrm{D}_\xi F'_{S^{d-1}}(M_0\rho_t)$. Furthermore, v_t^S is a tangent vector field of ρ_t with respect to the Wasserstein metric.

The proof of Lemma 5.20 can be found in Appendix C.3.1. In physical terms, we have the standard advection term $\operatorname{div}(v_t^S\rho_t)$ that is moving the particles along the vector field v_t^S in the same fashion as (5.8). We also have a reaction term $2F'_{S^{d-1}}(M_0\rho_t)$ that is removing mass in suboptimal regions of the space where $F'_{S^{d-1}}$ is negative, see Lemma 5.9.

Furthermore, we have a version of Gronwall inequality for the Wasserstein distance between two absolutely continuous curves and their tangent vector fields.

Lemma 5.21. *Let M be a complete Riemannian manifold, and let $m^j \colon [0,\infty) \to \mathcal{P}_2(M)$, $j = 1,2$, be absolutely continuous curves in the space of probability measures over M. Let v_t^j be a tangent vector field of m_t^j with respect to the Wasserstein metric on M. Then, for L^1-a.e. $t \geq 0$,*

$$\frac{\mathrm{d}}{\mathrm{d}t} W_2^2\left(m_t^1, m_t^2\right) \leq 2\int_{M^2} \Big(\big\langle \dot\gamma_{p_1,p_2}(1),\, v_t^2(p_2)\big\rangle - \big\langle \dot\gamma_{p_1,p_2}(0),\, v_t^1(p_1)\big\rangle\Big) \mathrm{d}\tau_t(p_1,p_2) \tag{5.30}$$

for $\tau_t \in \Gamma_o(m_t^1, m_t^2)$ and $\gamma_{p_1,p_2} \colon [0,1] \to M$ a geodesic joining p_1 and p_2. Moreover, if the geodesics satisfy $\sup_{t\in[0,1]} \|\ddot\gamma_{p_1,p_2}(t)\| \leq d_M(p_1,p_2)^2$, v_t^1 is Lip_t^1–Lipschitz, and v_t^2, resp. $v_t^1 - v_t^2$, is uniformly bounded by $\|v_t^2\|_\infty$, resp. $\|v_t^2 - v_t^1\|_\infty$, then

$$\frac{\mathrm{d}}{\mathrm{d}t} W_2^2\left(m_t^1, m_t^2\right) \leq 2\left(\left\|v_t^2\right\|_\infty + \mathrm{Lip}_t^1\right) W_2^2\left(m_t^1, m_t^2\right) + 2\left\|v_t^2 - v_t^1\right\|_\infty W_2\left(m_t^1, m_t^2\right).$$

The proof of Lemma 5.21 can be found in Appendix C.3.2.

5.4.3 Generalization bound: main result

Define, for a function $f\colon S^{d-1} \to \mathbb{R}$,

$$\|f\|_{C^2} \coloneqq \sup_{\xi \in S^{d-1}} \|\mathrm{D}_\xi f\|_\infty + \mathrm{Lip}_{\mathrm{D}f}, \tag{5.31}$$

where $\mathrm{Lip}_{\mathrm{D}f}$ is the Lipschitz constant of the derivative $D.f$. We now prove uniform stability of the Wasserstein gradient flow.

Proposition 5.22. *Under Assumption 5.17, $\mathcal{A}^T_{\mathrm{wgf}}$ has uniform stability β_T, where*

$$\beta_T = \exp\left(\int_0^T C_1(t)\mathrm{d}t\right) C_2(T) \int_0^T C_2(t)\mathrm{d}t \tag{5.32}$$

and

$$\begin{aligned} C_1(t) &\coloneqq C_\ell R_{\mathrm{emp}}(h(\mu_t), \mathcal{S}_n)^{1/2} \|\phi\|_{C^2} + \rho_n \|V\|_{C^2} + \|\ell\|_{C^2} \|\mathrm{D}\phi\|^2_{S^{d-1},\infty} \\ C_2(t) &\coloneqq C_\ell \|\mathrm{D}\phi\|_{S^{d-1},\infty} \sup_{(x,y)\in \mathrm{supp}(\mathcal{D})} \ell\left(\bar{y}(x,\mu_t),\, y\right)^{1/2}. \end{aligned} \tag{5.33}$$

Proof. Fix a sample set $\mathcal{S}_n = \{(x_i, y_i)\colon i = 1, \dots, n\}$ and let $\mu\colon \mathbb{R}_+ \to \mathcal{P}_2(\Theta)$ be the Wasserstein gradient flow for the functional $F_n(\,\cdot\,, \mathcal{S}_n)$, defined in (5.25). That means, $\mathcal{A}^T_{\mathrm{wgf}}(\mathcal{S}_n) = \mu_T$. For $i = 1, \dots, n$, the objective functional when removing the i^{th} sample (x_i, y_i) from the sample set $\mathcal{S}_n$ is denoted by $\mu \mapsto F_n(\mu, \mathcal{S}_n^{\setminus i}) = R_{\mathrm{emp}}(\mu, \mathcal{S}_n^{\setminus i}) + \rho_n \int_\Theta V(\theta)\mathrm{d}\mu(\theta)$. Let $\mu^{\setminus i}$ be the Wasserstein gradient flow for the functional $F_n(\,\cdot\,, \mathcal{S}_n^{\setminus i})$, so that $\mathcal{A}^T_{\mathrm{wgf}}(\mathcal{S}_n^{\setminus i}) = \mu_T^{\setminus i}$. Denote as well $\bar{y}(x, \mu) \coloneqq \int_\Theta \phi(\,\cdot\,, x)\mathrm{d}\mu = \int_{S^{d-1}} \phi(\,\cdot\,, x)\mathrm{d}h(\mu)$ the output of the mean-field map for the input $x \in \mathcal{X}$ and measure $\mu \in \mathcal{M}_+(\Theta)$.
Step 1: We first derive an explicit formula for the tangent vector field v_t^S of $h(\mu_t)$. Recall that the loss functional $L_{\mathcal{S}_n}\colon L^2(\mathcal{X}, \mathcal{Y}) \to \mathbb{R}$ is defined by $L(f) = \frac{1}{n}\sum_{i=1}^n \ell(f(x_i), y_i)$. From Lemma 5.20, we have for $\xi \in S^{d-1}$:

$$\begin{aligned} v_t^S(\xi) &= -\mathrm{D}_\xi F_n'(h(\mu_t), \mathcal{S}_n)(\xi) \\ &= -\left\langle \mathrm{d}L_{\mathcal{S}_n}\left(\int_{S^{d-1}} \Phi \mathrm{d}h(\mu_t)\right), \mathrm{D}_\xi \Phi(\xi)\right\rangle - \rho_n \mathrm{D}_\xi V(\xi) \\ &= -\frac{1}{n}\sum_{i=1}^n \frac{\partial \ell}{\partial y_1}\Big(\bar{y}(x_i, \mu_t),\, y_i\Big)\mathrm{D}_\xi\, \phi(\xi, x_i) - \rho_n \mathrm{D}_\xi V(\xi). \end{aligned} \tag{5.34}$$

Similarly, the tangent vector field $v_t^{S,\setminus i}$ of $h\big(\mu_t^{\setminus i}\big)$ is given by

$$v_t^{S,\setminus i}(\xi) = -\frac{1}{n}\sum_{i'\neq i} \frac{\partial \ell}{\partial y_1}\left(\bar{y}\left(x_{i'}, \mu_t^{\setminus i}\right),\, y_{i'}\right) \mathrm{D}_\xi\, \phi(\xi, x_{i'}) - \rho_n \mathrm{D}_\xi V(\xi). \tag{5.35}$$

Step 2: We now verify the conditions of the second part of Lemma 5.21. First, geodesics $\gamma_{p,v}$ on S^{d-1} with initial position and speed (p, v) satisfy $\ddot{\gamma}_{p,v} = -\|v\|^2 \gamma_{p,v}$, so $\|\gamma_{p_1,p_2}(t)\| = d_M(p_1, p_2)^2$, for each $t \in [0, 1]$. Next, Assumption 5.17 together with (5.34) ensures that v_t^S is Lip_t^S–Lipschitz continuous, with

$$\begin{aligned}\mathrm{Lip}_t^S &= \frac{1}{n}\sum_{i=1}^{n}\left\|\frac{\partial \ell}{\partial y_1}\left(\overline{y}(x_i, \mu_t),\, y_i\right)\right\| \mathrm{Lip}_{\mathrm{D}\phi,\, S^{d-1}, \infty} + \rho_n \mathrm{Lip}_{\mathrm{D}V,\, S^{d-1}} \\ &\leq C_\ell R_{\mathrm{emp}}(h(\mu_t), \mathcal{S}_n)^{1/2} \mathrm{Lip}_{\mathrm{D}\phi,\, S^{d-1}, \infty} + \rho_n \mathrm{Lip}_{\mathrm{D}V,\, S^{d-1}}. \qquad (5.36)\end{aligned}$$

We also note from (5.35) that $v_t^{S,\setminus i}$ is uniformly bounded by

$$\begin{aligned}\left\|v_t^{S,\setminus i}\right\|_\infty &= \frac{1}{n}\sum_{i' \neq i}\left\|\frac{\partial \ell}{\partial y_1}\left(\overline{y}\left(x_{i'}, \mu_t^{\setminus i}\right),\, y_{i'}\right)\right\| \|\mathrm{D}\phi\|_{S^{d-1}, \infty} + \rho_n \|\mathrm{D}V\|_{S^{d-1}} \\ &\leq C_\ell R_{\mathrm{emp}}(h(\mu_t), \mathcal{S}_n)^{1/2} \|\mathrm{D}\phi\|_{S^{d-1}, \infty} + \rho_n \|\mathrm{D}V\|_{S^{d-1}}. \qquad (5.37)\end{aligned}$$

Furthermore, for $\xi \in S^{d-1}$, we estimate the difference

$$\begin{aligned}&\left\|v_t^S(\xi) - v_t^{S,\setminus i}(\xi)\right\| \\ &\leq \frac{1}{n}\left\|\frac{\partial \ell}{\partial y_1}\left(\overline{y}(x_i, \mu_t),\, y_i\right)\right\| \|\mathrm{D}\phi\|_{S^{d-1}, \infty} \\ &\quad + \frac{1}{n}\sum_{i' \neq i}\left\|\frac{\partial \ell}{\partial y_1}\left(\overline{y}\left(x_{i'}, \mu_t^{\setminus i}\right),\, y_{i'}\right) - \frac{\partial \ell}{\partial y_1}\left(\overline{y}(x_{i'}, \mu_t),\, y_{i'}\right)\right\| \|\mathrm{D}\phi\|_{S^{d-1}, \infty} \\ &\leq \frac{1}{n} C_\ell\, \ell\big(\overline{y}(x_i, \mu_t),\, y_i\big)^{1/2} \|\mathrm{D}\phi\|_{S^{d-1}, \infty} + \frac{1}{n}\sum_{i' \neq i}\|\ell\|_{C^2}\left\|\overline{y}\left(x_{i'}, \mu_t^{\setminus i} - \mu_t\right)\right\| \|\mathrm{D}\phi\|_{S^{d-1}, \infty}\end{aligned}$$

Therefore, by the dual formulation of the 1–Wasserstein distance, we deduce

$$\begin{aligned}\left\|v_t^S - v_t^{S,\setminus i}\right\|_\infty \leq & \frac{1}{n} C_\ell\, \ell\big(\overline{y}(x_i, \mu_t),\, y_i\big)^{1/2} \|\mathrm{D}\phi\|_{S^{d-1}, \infty} \\ &+ \|\ell\|_{C^2} \|\mathrm{D}\phi\|_{S^{d-1}, \infty}^2 W_1\left(h\left(\mu_t^{\setminus i}\right),\, h(\mu_t)\right) \qquad (5.38)\end{aligned}$$

Step 3: We combine (5.36), (5.37), and (5.38) together with Lemma 5.21 to get

$$\frac{\mathrm{d}}{\mathrm{d}t} W_2^2\left(h(\mu_t),\, h\left(\mu_t^{\setminus i}\right)\right) \leq 2C_1(t) W_2^2\left(h(\mu_t),\, h\left(\mu_t^{\setminus i}\right)\right) + \frac{2}{n} C_2(t) W_2\left(h(\mu_t),\, h\left(\mu_t^{\setminus i}\right)\right),$$

where C_1 and C_2 are defined in (5.33). We can thus apply Lemma D.5, a variant of Gronwall lemma, to deduce the following bound on the Wasserstein distance between μ_T and $\mu_T^{\setminus i}$.

$$W_2\left(h(\mu_T),\, h\left(\mu_T^{\setminus i}\right)\right) \leq \frac{1}{n}\exp\left(\int_0^T C_1(t)\mathrm{d}t\right)\int_0^T C_2(t)\mathrm{d}t. \qquad (5.39)$$

Step 5: Finally, we get that for each $(x, y) \in \operatorname{supp}(\mathcal{D})$,

$$\begin{aligned}\left|\ell\Big(\bar{y}(x, \mu_T), y\Big) - \ell\Big(\bar{y}\big(x, \mu_T^{\setminus i}\big), y\Big)\right| &\leq \left\|\frac{\partial \ell}{\partial y_1}\big(\bar{y}(x, \mu_T), y\big)\right\| \left\|\bar{y}\big(x, \mu_T - \mu_T^{\setminus i}\big)\right\| \\ &\leq C_\ell\, \ell\big(\bar{y}(x, \mu_T), y\big)^{1/2} \left\|\mathrm{D}\phi\right\|_{S^{d-1}, \infty} W_1\left(h(\mu_T),\, h\big(\mu_T^{\setminus i}\big)\right) \\ &\leq C_2(T) W_2\left(h(\mu_T),\, h\big(\mu_T^{\setminus i}\big)\right).\end{aligned}$$

The result follows by definition. □

We can now apply Proposition 5.19 to compute an exponential generalization bound for the Wasserstein gradient flow alogrithm.

Theorem 5.23. *Let $\delta \in (0, 1)$. Under Assumption 5.17, with probability $1 - \delta$ over the choice of the dataset $\mathcal{S}_n$, we have*

$$R\left(\mathcal{A}_{\mathrm{wgf}}^T(\mathcal{S}_n)\right) \leq R_{\mathrm{emp}}\left(\mathcal{A}_{\mathrm{wgf}}^T(\mathcal{S}_n), \mathcal{S}_n\right) + \frac{2\beta_T}{n} + (4\beta_T + B)\sqrt{\frac{\log 1/\delta}{2n}}, \tag{5.40}$$

where β_T is defined in (5.32) and

$$B = 2\left\|\ell\right\|_{C^2}\left(\left\|\phi\right\|_{S^{d-1}, \infty}^2 \left(\int_\Theta \left\|u\right\|^2 \mathrm{d}\mu_0(u)\right)^2 + \sup_{y \sim \mathcal{D}_Y} \left\|y\right\|^2\right).$$

Proof. In order to apply Proposition 5.19, we need to compute the uniform bound B on the loss function. We first have

$$\ell(y', y) \leq \ell(y, y) + \frac{\partial \ell}{\partial y_1}(y, y)^\top (y' - y) + \left\|\ell\right\|_{C^2} \left\|y' - y\right\|^2$$

Assumption 5.17 *(iii)* states that $\ell(y, y) = 0$, as well as $\|\partial \ell / \partial y_1(y, y)\| \leq C_\ell \ell(y, y)^{1/2} = 0$. Therefore, $\ell(y', y) \leq \left\|\ell\right\|_{C^2} \left\|y' - y\right\|^2$. Now,

$$\begin{aligned}\int_\Theta \phi(\,\cdot\,, x) \mathrm{d}\mathcal{A}_{\mathrm{wgf}}^T(\mathcal{S}_n) &= \int_{S^{d-1}} \phi(\,\cdot\,, x) \mathrm{d}h(\mathcal{A}_{\mathrm{wgf}}^T(\mathcal{S}_n)) \\ &\leq \left\|\phi\right\|_{S^{d-1}, \infty} \int_\Theta \left\|u\right\|^2 \mathrm{d}\mu_0(u)\end{aligned}$$

The last step holds by Lemma 5.20. We conclude that

$$\ell\left(\int_\Theta \phi(\,\cdot\,, x) \mathrm{d}\mathcal{A}_{\mathrm{wgf}}^T(\mathcal{S}_n),\, y\right) \leq 2\left\|\ell\right\|_{C^2}\left(\left\|\phi\right\|_{S^{d-1}, \infty}^2 \left(\int_\Theta \left\|u\right\|^2 \mathrm{d}\mu_0(u)\right)^2 + \sup_{y \sim \mathcal{D}_Y} \left\|y\right\|^2\right)$$

□

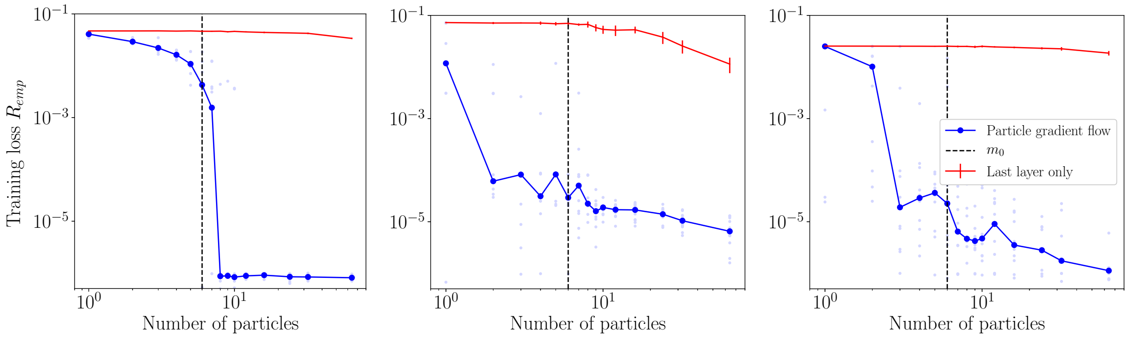

Figure 5.2: Final training loss against the number of particles m, with $m_0 = 6$. The light blue points are 10 individual runs, the solid blue line is their median and the red curve is only optimizing the last layer. Left: $d_0 = 64$, $K = 2$. Middle: $d_0 = 8$, $d_h = 8$, $K = 3$. Right: $d_0 = 4$, $d_h = 4$, $K = 4$.

We observe from (5.32) that β_T depends exponentially on the horizon (or number of iterations) T. This is in contrast with convex losses, that show a linear dependence on the number of iterations for gradient descent [69].

Now, in a sparse optimization setting, we have linear convergence of gradient flow [34]: T does not depend on n or d, which means that our bound (5.40) does not suffer from the curse of dimensionality, despite having weak assumptions on the data distribution. However, this bound is not practical as approximating the Wasserstein gradient flow with m particles induces an error that scales like $m^{1/d}$. The results are in line with [12]: when no assumption on the data manifold is made, the generalization bound scales like $n^{1/d}$.

5.5 Numerical experiments

In this section, we empirically study the previous abstract theory on specific examples by simulating the particle gradient flow using discrete SGD steps with gradients for each model computed with respect to the average prediction. We show on synthetic experiments that the particle gradient flow converges to an optimal set of parameters. The experiments on CIFAR-10 classification demonstrate favourable generalization properties.

5.5.1 Particle complexity for convergence of multi-layer ReLU networks

We numerically investigate the particle complexity required for the convergence of the particle gradient flow to a global minimum. To do so, we generate synthetic data and a true neural network that achieves zero loss on the data as follows. We

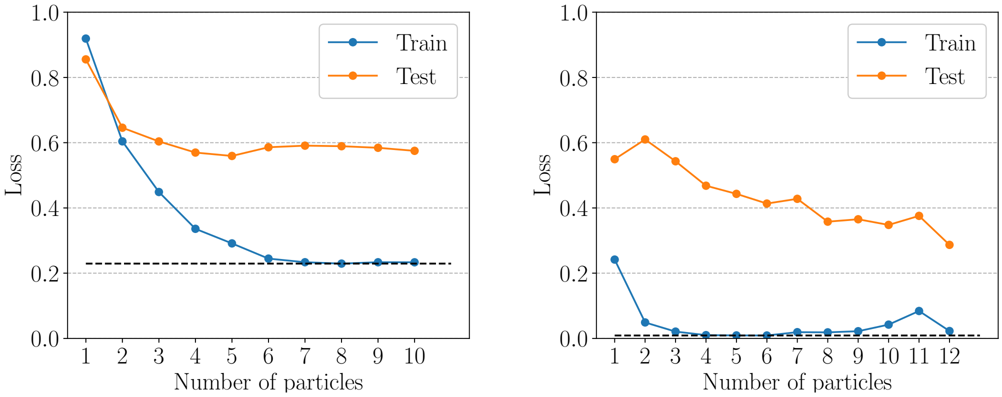

Figure 5.3: Particle gradient flow on AlexNetSmall and VGG11 tested on CIFAR-10. Left: AlexNetSmall on CIFAR-10. Right: VGG11 (11 layers)

generate the input data from the uniform distribution $x \in [-1,1]^{d_0}$, the parameters $(\overline{\theta}_j)_{j=1}^{m_0}$ are fixed weights and biases of a K-layer neural network, and the output labels $y = \frac{1}{m_0}\sum_{j=1}^{m_0} \phi(\overline{\theta}_j, x)$, where m_0 is the true number of particles. We simulate the gradient flow using a mixture of m particles with the mean squared loss ℓ using $10,000$ gradient updates with early stopping if the loss has not improved in the last $1,000$ updates or if the loss dropped below 10^{-6}. We compare the particle gradient flow with randomly initializing m particles $(\widetilde{\theta}_j)_{j=1}^{m}$ and only optimizing the last layer containing the weights $(w_j)_{j=1}^{m}$, so that the input-output map $x \mapsto \sum_{j=1}^{m} w_j \phi(\widetilde{\theta}_j, x)$ is linear in w. We see in Figure 5.2 that the effect of depth is two-fold. First, as the depth increases the individual networks ϕ become more expressive and the gradient flow converges to a lower loss even with less than m_0 particles. On the other hand, the networks become harder to train to global convergence, requiring more particles or longer training to achieve the target loss of 10^{-6}.

5.5.2 Convergence of the particle gradient flow of deep convolutional networks

We empirically explore the convergence of particle gradient flow on ensembles of two different architectures, VGG11 [140] (11 layers) and a smaller modification of an AlexNet [92] that we call AlexNetSmall (6 layers) on CIFAR-10.

Figure 5.3 shows the final loss of models after 250 epochs of training on CIFAR-10. We see that the training loss of both models is improved with increasing the number of particles until it converges to a maximum at $\widetilde{m} = 8$ for AlexNetSmall and $\widetilde{m} = 4$ for VGG11, beyond which the training capabilities do not improve.

Surprisingly, we do not observe overfitting behaviour after increasing the number of particles beyond $\widetilde{m}$. For AlexNetSmall the test loss stays roughly constant for $m \geq 5$. For VGG11, we observe an improvement in test loss even in a regime where the training loss has saturated. We believe that this suggests favorable generalization properties of average ensembles trained with particle gradient flow.

Bibliography

[1] Ralph Abraham and Joel Robbin. *Transversal mappings and flows.* W.A. Benjamin, New York, USA, 1967.

[2] Ibrahim M Alabdulmohsin. Algorithmic stability and uniform generalization. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[3] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and Generalization in Overparameterized Neural Networks, Going Beyond Two Layers. In *Advances in Neural Information Processing Systems*, pages 6158–6169, 2019.

[4] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A Convergence Theory for Deep Learning via Over-Parameterization. In *Proceedings of Machine Learning Research*, volume 97, pages 242–252, 2019.

[5] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient Flows in Metric Spaces and in the Space of Probability Measures*. Lectures in Mathematics, ETH Zürich, Zurich, Switzerland, 2008.

[6] Dyego Araújo, Roberto I. Oliveira, and Daniel Yukimura. A mean-field limit for certain deep neural networks, 2019.

[7] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A Convergence Analysis of Gradient Descent for Deep Linear Neural Networks. In *7th International Conference on Learning Representations (ICLR)*, 2019.

[8] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit Regularization in Deep Matrix Factorization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[9] Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *Proceedings of the 36th International Conference on*

Machine Learning, volume 97 of *Proceedings of Machine Learning Research*, pages 322–332. PMLR, 2019.

[10] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[11] Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 254–263. PMLR, 2018.

[12] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18(19):1–53, 2017.

[13] Thomas Bachlechner, Bodhisattwa Prasad Majumder, Henry Mao, Gary Cottrell, and Julian McAuley. ReZero is all you need: fast convergence at large depth. In *Proceedings of Machine Learning Research*, volume 161, pages 1352–1361. PMLR, 2021.

[14] David Barrett and Benoit Dherin. Implicit gradient regularization. In *International Conference on Learning Representations*, 2021.

[15] Peter Bartlett, Dave Helmbold, and Philip Long. Gradient descent with identity initialization efficiently learns positive definite linear transformations by deep residual networks. In *International Conference on Machine Learning (ICML)*, pages 521–530. PMLR, 2018.

[16] Peter L. Bartlett, Steven N. Evans, and Philip M. Long. Representing smooth functions as compositions of near-identity functions with implications for deep network optimization, 2018.

[17] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

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

[19] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

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

[21] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, page 437–478. Springer, 2012.

[22] Leon Bottou. *Online Algorithms and Stochastic Approximations*. Cambridge University Press, Cambridge, UK, 1998.

[23] Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[24] Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

[25] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[26] Alan J Bray and David S Dean. Statistics of critical points of gaussian fields on large-dimensional spaces. *Physical Review Letters*, 98(15):150–201, 2007.

[27] Yuan Cao and Quanquan Gu. Generalization Bounds of Stochastic Gradient Descent for Wide and Deep Neural Networks. In *Advances in Neural Information Processing Systems (NeuIPS), 2019*, 2019.

[28] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems 31*, pages 6571–6583, 2018.

[29] Zixiang Chen, Yuan Cao, Quanquan Gu, and Tong Zhang. A generalized neural tangent kernel analysis for two-layer neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 13363–13373, 2020.

[30] Xiang Cheng, Dong Yin, Peter Bartlett, and Michael Jordan. Stochastic gradient and Langevin processes. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 1810–1819, 2020.

[31] Lenaic Chizat and Francis Bach. On the Global Convergence of Gradient Descent for Over-parameterized Models using Optimal Transport. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[32] Lenaic Chizat and Francis Bach. Implicit Bias of Gradient Descent for Wide Two-layer Neural Networks Trained with the Logistic Loss, February 2020.

[33] Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On Lazy Training in Differentiable Programming. In *Advances in Neural Information Processing Systems*, 2019.

[34] Lénaïc Chizat. Sparse Optimization on Measures with Over-parameterized Gradient Descent, July 2019.

[35] Youngmin Cho and Lawrence Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, volume 22, 2009.

[36] Alain-Sam Cohen, Rama Cont, Alain Rossier, and Renyuan Xu. Scaling Properties of Deep Residual Networks. In *Proceedings of the 38th International Conference on Machine Learning*, pages 2039–2048, 2021.

[37] Alain-Sam Cohen, Rama Cont, Alain Rossier, and Renyuan Xu. Scaling properties of deep residual networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2039–2048. PMLR, 18–24 Jul 2021.

[38] George V. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:303–314, 1989.

[39] Amit Daniely, Roy Frostig, and Yoram Singer. Toward Deeper Understanding of Neural Networks: The Power of Initialization and a Dual View on Expressivity. In *Advances in Neural Information Processing Systems 29*, pages 2253–2261, 2016.

[40] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[41] Luc Devroye, László Györfi, and Gábor Lugosi. *A Probabilistic Theory of Pattern Recognition*, volume 1. Springer, New York, USA, 1996.

[42] Hans M. Dietz. On the solution of matrix-valued linear stochastic differential equations driven by semimartingales. *Stochastics: An International Journal of Probability and Stochastic Processes*, 34:127–147, 1991.

[43] Simon Du and Wei Hu. Width Provably Matters in Optimization for Deep Linear Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 1655–1664, 2019.

[44] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient Descent Finds Global Minima of Deep Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*, pages 1675–1685, 2019.

[45] Simon S. Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations*, 2019.

[46] Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented Neural ODEs. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[47] Weinan E, Jiequn Han, and Qianxiao Li. A mean-field optimal control formulation of deep learning. *Research in the Mathematical Sciences*, 6(1):1–41, 2019.

[48] Weinan E, Chao Ma, Qingcan Wang, and Lei Wu. Analysis of the gradient descent algorithm for a deep neural network model with skip-connections. *CoRR*, abs/1904.05263, 2019.

[49] Weinan E, Chao Ma, Qingcan Wang, and Lei Wu. Analysis of the Gradient Descent Algorithm for a Deep Neural Network Model with Skip-connections. *arXiv preprint arXiv:1904.05263*, 2019.

[50] Ronen Eldan and Ohad Shamir. The Power of Depth for Feedforward Neural Networks. In *29th Annual Conference on Learning Theory*, pages 907–940. PMLR, 2016.

[51] Michel Emery. Equations différentielles stochastiques lipschitziennes: étude de la stabilité. *Séminaire de probabilités (Strasbourg)*, 13:281–293, 1979.

[52] Vitaly Feldman and Jan Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. In Alina Beygelzimer and Daniel Hsu, editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 1270–1279. PMLR, 2019.

[53] Gerald B Folland. *Real Analysis: Modern Techniques and Their Applications*. John Wiley & Sons, 1999.

[54] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

[55] C. Daniel Freeman and Joan Bruna. Topology and geometry of half-rectified network optimization. In *International Conference on Learning Representations*, 2017.

[56] Spencer Frei, Yuan Cao, and Quanquan Gu. Algorithm-dependent generalization bounds for overparameterized deep residual networks. *CoRR*, abs/1910.02934, 2019.

[57] Thomas O. Gallouët and Léonard Monsaingeon. A jko splitting scheme for kantorovich–fisher–rao gradient flows. *SIAM Journal on Mathematical Analysis*, 49(2):1100–1130, 2017.

[58] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit Regularization of Discrete Gradient Dynamics in Linear Neural Networks. In *Advances in Neural Information Processing Systems 32*, pages 3202–3211, 2019.

[59] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of Machine Learning Research*, volume 9, pages 249–256, 2010.

[60] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[61] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[62] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[63] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour, 2018.

[64] Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Characterizing implicit bias in terms of optimization geometry. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 1832–1841, 2018.

[65] Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit Bias of Gradient Descent on Linear Convolutional Networks. In *Advances in Neural Information Processing Systems 31*, pages 9461–9471, 2018.

[66] Suriya Gunasekar, Blake Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, 2017.

[67] Eldad Haber and Lars Ruthotto. Stable architectures for deep neural networks. *Inverse Problems*, 34(1), 2018.

[68] Moritz Hardt and Tengyu Ma. Identity Matters in Deep Learning. In *5th International Conference on Learning Representations (ICLR)*, 2017.

[69] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1225–1234, New York, New York, USA, 20–22 Jun 2016. PMLR.

[70] Fengxiang He, Tongliang Liu, and Dacheng Tao. Control batch size and learning rate to generalize well: Theoretical and empirical evidence. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[71] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[72] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity Mappings in Deep Residual Networks. In *European Conference on Computer Vision*, pages 630–645, 2016.

[73] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 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*, pages 770–778. IEEE Computer Society, 2016.

[74] Jahn Heymann, Lukas Drude, and Reinhold Haeb-Umbach. Wide Residual BLSTM Network with Discriminative Speaker Adaptation for Robust Speech Recognition. In *Computer Speech and Language*, 2016.

[75] Desmond J Higham, Xuerong Mao, and Andrew M Stuart. Strong convergence of euler-type methods for nonlinear stochastic differential equations. *SIAM Journal on Numerical Analysis*, 40(3):1041–1063, 2002.

[76] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6:107–116, 04 1998.

[77] Kaitong Hu, Zhenjie Ren, David Siska, and Lukasz Szpruch. Mean-Field Langevin Dynamics and Energy Landscape of Neural Networks. *arXiv:1905.07769*, May 2019.

[78] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Weinberger. Deep Networks with Stochastic Depth. In *European Conference on Computer Vision*, 2016.

[79] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, 1:448–456, 2015.

[80] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *CoRR*, abs/1502.03167, 2015.

[81] Kiyosi Itô. Stochastic integral. *Proc. Imp. Acad.*, 20(8):519–524, 1944.

[82] Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. In *Advances in Neural Information Processing Systems*, volume 31, pages 8571–8580, 2018.

[83] Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks. In *International Conference on Learning Representations*, 2019.

[84] Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks, 2019.

[85] Yiding Jiang*, Behnam Neyshabur*, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

[86] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

[87] Kenji Kawaguchi. Deep Learning without Poor Local Minima. In *Advances in Neural Information Processing Systems 29*, pages 586–594, 2016.

[88] Kenji Kawaguchi, Leslie Pack Kaelbling, and Yoshua Bengio. Generalization in deep learning. *Mathematical Aspects of Deep Learning*, pages 112–148, 2022.

[89] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.

[90] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[91] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Citeseer*, 2009.

[92] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, pages 1097—-1105, 2012.

[93] Thomas Laurent and James von Brecht. Deep Linear Networks with Arbitrary Loss: All Local Minima Are Global. In *Proceedings of Machine Learning Research*, volume 80, 2018.

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

[95] Jaehoon Lee, Lechao Xiao, Samuel S Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent*. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12), 2020.

[96] Jason D. Lee, Max Simchowitz, Michael I. Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1246–1257, 2016.

[97] Qianxiao Li, Cheng Tai, and Weinan E. Stochastic modified equations and adaptive stochastic gradient algorithms. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 2101–2110, 2017.

[98] Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David K. Duvenaud. Scalable Gradients and Variational Inference for Stochastic Differential Equations. In *Proceedings of The 2nd Symposium on Advances in Approximate Bayesian Inference*, volume 118 of *Proceedings of Machine Learning Research*, pages 1–28. PMLR, 2020.

[99] Zhiyuan Li, Sadhika Malladi, and Sanjeev Arora. On the validity of modeling sgd with stochastic differential equations (sdes). In *Advances in Neural Information Processing Systems*, volume 34, pages 12712–12725, 2021.

[100] Shiyu Liang and R. Srikant. Why deep neural networks for function approximation? In *International Conference on Learning Representations*, 2017.

[101] Matthias Liero, Alexander Mielke, and Giuseppe Savaré. Optimal transport in competition with reaction: The hellinger–kantorovich distance and geodesic curves. *SIAM Journal on Mathematical Analysis*, 48(4):2869–2911, 2016.

[102] Y. Liu and X. Yao. Ensemble learning via negative correlation. *Neural Networks*, 12(10):1399–1404, dec 1999.

[103] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with restarts. *CoRR*, abs/1608.03983, 2016.

[104] Haihao Lu and Kenji Kawaguchi. Depth Creates No Bad Local Minima, 2017.

[105] Yiping Lu, Chao Ma, Yulong Lu, Jianfeng Lu, and Lexing Ying. A Mean-field Analysis of Deep ResNet and Beyond: Towards Provable Optimization Via Overparameterization From Depth, March 2020.

[106] Yiping Lu, Chao Ma, Yulong Lu, Jianfeng Lu, and Lexing Ying. A mean field analysis of deep ResNet and beyond: Towards provably optimization via overparameterization from depth. In *International Conference on Machine Learning*, pages 6426–6436. PMLR, 2020.

[107] Yiping Lu, Aoxiao Zhong, Quanzheng Li, and Bin Dong. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In *35th International Conference on Machine Learning, ICML*, pages 5181–5190, 2018.

[108] Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 3325–3334, 2018.

[109] Stéphane Mallat. Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065), 2016.

[110] Stephan Mandt, Matthew D. Hoffman, and David M. Blei. Stochastic gradient descent as approximate bayesian inference. *Journal of Machine Learning Research*, 18(134):1–35, 2017.

[111] Alexander G. de G. Matthews, Mark Rowland, Jiri Hron, Richard E. Turner, and Zoubin Ghahramani. Gaussian process behaviour in wide deep neural networks, 2018.

[112] Song Mei and Andrea Montanari. The Generalization Error of Random Features Regression: Precise Asymptotics and Double Descent Curve, 2019.

[113] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4):667–766, 2022.

[114] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

[115] H. N. Mhaskar. Neural networks for optimal approximation of smooth and analytic functions. *Neural Computation*, 8(1):164–177, 1996.

[116] Vaishnavh Nagarajan and J. Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[117] Behnam Neyshabur, Russ R Salakhutdinov, and Nati Srebro. Path-sgd: Path-normalized optimization in deep neural networks. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pages 2422–2430, 2015.

[118] Behnam Neyshabur, Ryota Tomioka, Ruslan Salakhutdinov, and Nathan Srebro. Geometry of Optimization and Implicit Regularization in Deep Learning, May 2017.

[119] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning. *arXiv preprint arXiv:1412.6614*, 2014.

[120] Quynh Nguyen and Matthias Hein. The loss surface of deep and wide neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, page 2603–2612, 2017.

[121] Atsushi Nitand and Taiji Suzuki. Stochastic Particle Gradient Descent for Infinite Ensembles. *arXiv:1712.05438*, December 2017.

[122] Roman Novak, Lechao Xiao, Yasaman Bahri, Jaehoon Lee, Greg Yang, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-dickstein. Bayesian deep convolutional networks with many channels are gaussian processes. In *International Conference on Learning Representations*, 2019.

[123] Katharina Ott, Prateek Katiyar, Philipp Hennig, and Michael Tiemann. ResNet After All: Neural ODEs and Their Numerical Solution. In *International Conference on Learning Representations*, 2021.

[124] Stefano Peluchetti and Stefano Favaro. Infinitely deep neural networks as diffusion processes. In *Intl Conference on Artificial Intelligence and Statistics*, pages 1126–1136. PMLR, 2020.

[125] Jeffrey Pennington, Samuel Schoenholz, and Surya Ganguli. The emergence of spectral universality in deep networks. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 1924–1932, 2018.

[126] Henning Petzka and Cristian Sminchisescu. Non-attracting Regions of Local Minima in Deep and Wide Neural Networks. *Journal of Machine Learning Research*, 22(143):1–34, 2021.

[127] Eckhard Platen and Nicola Bruti-Liberati. *Numerical solution of stochastic differential equations with jumps in finance*, volume 64. Springer Science & Business Media, 2010.

[128] B. T. Polyak. Gradient methods for minimizing functionals. *U.S.S.R. Comput. Math. Math. Phys.*, 3:864–878, 06 1963.

[129] Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

[130] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems 20*, pages 1177–1184, 2008.

[131] Daniel Revuz and Marc Yor. *Continuous martingales and Brownian motion*. Springer, 2013.

[132] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[133] Grant Rotskoff and Eric Vanden-Eijnden. Parameters as interacting particles: long time convergence and asymptotic error scaling of neural networks. In *Advances in Neural Information Processing Systems (NeurIPS 2018)*, pages 7146–7155, 2018.

[134] Alessandro Rudi and Lorenzo Rosasco. Generalization properties of learning with random features. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, pages 3215–3225, 2017.

[135] Walter Rudin. *Principles of Mathematical Analysis*. McGraw-Hill Publishing Company, New York City, USA, 1964.

[136] Daniel Russo and James Zou. Controlling bias in adaptive data analysis using information theory. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 1232–1240. PMLR, 2016.

[137] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(4):e1249, 2018.

[138] Michael E. Sander, Pierre Ablin, and Gabriel Peyré. Do Residual Neural Networks discretize Neural Ordinary Differential Equations? In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[139] Hanie Sedghi, Vineet Gupta, and Philip M. Long. The singular values of convolutional layers. *CoRR*, abs/1805.10408, 2018.

[140] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations (ICLR 2015)*, 2015.

[141] Justin Sirignano and Konstantinos Spiliopoulos. Mean Field Analysis of Neural Networks: A Law of Large Numbers. *SIAM Journal on Applied Mathematics*, 80(2):725–752, jan 2020.

[142] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of deep neural networks. *Mathematics of Operations Research*, 47(1):120–152, 2022.

[143] Samuel L Smith, Benoit Dherin, David Barrett, and Soham De. On the origin of implicit regularization in stochastic gradient descent. In *International Conference on Learning Representations*, 2021.

[144] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 2256–2265, 2015.

[145] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[146] Ruoyu Sun, Dawei Li, Shiyu Liang, Tian Ding, and Rayadurgam Srikant. The Global Landscape of Neural Networks: An Overview. *IEEE Signal Processing Magazine*, 37(5):95–108, 2020.

[147] Matus Telgarsky. Representation Benefits of Deep Feedforward Networks, 2015.

[148] Matthew Thorpe and Yves van Gennip. Deep Limits of Residual Neural Networks. *Res. Math Sci.*, 10(6), 2023.

[149] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[150] Andreas Veit, Michael J Wilber, and Serge Belongie. Residual networks behave like ensembles of relatively shallow networks. In *Advances in Neural Information Processing Systems 29 (NIPS 2016)*, pages 550–558, 2016.

[151] Cédric Villani. *Topics in Optimal Transportation*, volume Graduate Studies in Mathematics 58. American Mathematical Society, Providence, Rhode Island, 2003.

[152] Lei Wu, Qingcan Wang, and Chao Ma. Global Convergence of Gradient Descent for Deep Linear Residual Networks. In *Advances in Neural Information Processing Systems 32*, pages 13389–13398, 2019.

[153] Lei Wu, Qingcan Wang, and Chao Ma. Global Convergence of Gradient Descent for Deep Linear Residual Networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[154] Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[155] Huan Xu and Shie Mannor. Robustness and generalization. *Machine Learning*, 86(3):391–423, 2012.

[156] Mengjia Xu, Akshay Rangamani, Qianli Liao, Tomer Galanti, and Tomaso Poggio. Dynamics in deep classifiers trained with the square loss: Normalization, low rank, neural collapse, and generalization bounds. *Research*, 6:0024, 2023.

[157] Liqing Yan. Right and left matrix-valued stochastic exponentials and explicit solutions to systems of sdes. *Stochastic Analysis and Applications*, 30:1:160–173, 2012.

[158] Ge Yang and Samuel Schoenholz. Mean Field Residual Networks: On the Edge of Chaos. In *Advances in Neural Information Processing Systems 30*, pages 7103–7114, 2017.

[159] Dmitry Yarotsky. Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114, 2017.

[160] Chulhee Yun, Shankar Krishnan, and Hossein Mobahi. A Unifying View on Implicit Bias in Training Linear Neural Networks. *arXiv preprint arXiv:2010.02501*, 2020.

[161] Sergey Zagoruyko and Nikos Komodakis. Wide Residual Networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press, 2016.

[162] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding Deep Learning (still) Requires Rethinking Generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[163] Huishuai Zhang, Da Yu, Mingyang Yi, Wei Chen, and Tie-Yan Liu. Convergence Theory of Learning Over-parameterized ResNet: A Full Characterization, 2019.

[164] Difan Zou and Quanquan Gu. An Improved Analysis of Training Over-parameterized Deep Neural Networks. In *Advances in Neural Information Processing Systems*, pages 2055–2064, 2019.

[165] Difan Zou, Philip M Long, and Quanquan Gu. On the Global Convergence of Training Deep Linear ResNets. *8th International Conference on Learning Representations (ICLR)*, 2020.

Appendix A

Hyperparameters

We provide in Table A.1 the training hyperparameters used in our numerical experiments. In Table A.2, we give a short description of each hyperparameter. For the convolutional architecture, we also use a momentum of 0.9, a weight decay of 0.0005 and a cosine annealing learning rate scheduler [103].

Table A.1: Training hyperparameters.

Dataset	Layers	N	B	η	L_{min}	L_{max}	T_{max}	N_{epochs}	ϵ
Synthetic	Fully-connected	1,024	32	0.01	3	10,321	160	5	0.01
MNIST	Fully-connected	60,000	50	0.01	3	942	12,000	10	0.01
CIFAR-10	Convolutional	60,000	128	0.1	8	121	93,800	200	None

Table A.2: Description of the values in Table A.1. Note that $T_{\text{max}} = \left\lceil \frac{N}{B} \right\rceil N_{\text{epochs}}$.

Parameter	Description
N	number of training samples
B	minibatch size
η	learning rate
L_{min}	smallest network depth
L_{max}	largest network depth
T_{max}	max number of SGD updates
N_{epochs}	max number of epochs
ϵ	early stopping value

Appendix B

Technical results of Chapter 4

B.1 Gradient of the loss function with respect to parameters

Let $x, y \in \mathbb{R}^d$ and $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$. We want to compute the gradient of $\ell(y, \widehat{y}(x, \alpha^{(L)}))$ with respect to the network parameters $\{\alpha_k^{(L)} : k = 1, \ldots, L\}$. Fix $1 \le k \le L$ and $1 \le m, n \le d$. We first observe that

$$\frac{\partial \ell}{\partial \alpha_{k,mn}^{(L)}} \left(y, \widehat{y}(x, \alpha^{(L)})\right) = \nabla_{\widehat{y}}\, \ell\left(y, \widehat{y}\left(x, \alpha^{(L)}\right)\right)^\top \frac{\partial h_L^{(L)}}{\partial h_k^{(L)}} \frac{\partial h_k^{(L)}}{\partial \alpha_{k,mn}^{(L)}}.$$

By induction, we obtain

$$\begin{aligned} M_k^{(L)} := \frac{\partial h_L^{(L)}}{\partial h_k^{(L)}} &= \prod_{j=k+1}^{L} \frac{\partial h_j^{(L)}}{\partial h_{j-1}^{(L)}} \\ &= \prod_{j=k+1}^{L} \left(I_d + \delta_L \frac{\partial}{\partial h_{j-1}^{(L)}} \sigma_d \left(\alpha_j^{(L)} h_{j-1}^{(L)} \right) \right) \\ &= \prod_{j=k+1}^{L} \left(I_d + \delta_L \mathrm{diag} \left(\nabla \sigma_d \left(\alpha_j^{(L)} h_{j-1}^{(L)} \right) \right) \alpha_j^{(L)} \right). \end{aligned} \tag{B.1}$$

We also have

$$\frac{\partial h_k^{(L)}}{\partial \alpha_{k,mn}^{(L)}} = \delta_L \sigma' \left(\left(\alpha_k^{(L)} h_{k-1}^{(L)} \right)_m \right) h_{k-1,n}^{(L)} e_m \in \mathbb{R}^d.$$

Denote $\dot{\sigma}_{k,m}^{(L)} := \sigma' \left(\left(\alpha_k^{(L)} h_{k-1}^{(L)} \right)_m \right)$. Regrouping everything, we get

$$\frac{\partial \ell}{\partial \alpha_{k,mn}^{(L)}} = \delta_L\, h_{k-1,n}^{(L)}\, \dot{\sigma}_{k,m}^{(L)}\, \nabla_{\widehat{y}}\, \ell\left(y, \widehat{y}\left(x, \alpha^{(L)}\right)\right)^\top M_k^{(L)} e_m. \tag{B.2}$$

B.2 Boundedness of hidden states and Jacobians

This section contains two useful results for our analysis.

Lemma B.1. *Let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ and $c_A > 0$ such that $L \geq 5c_A$ and*

$$\left\|\alpha^{(L)}\right\|_{F,\infty} = \max_{k=1,\dots,L} \left\|\alpha_k^{(L)}\right\|_F \leq c_A L^{-1/2}.$$

Then, under Assumption 4.1 (i)–(ii), we have that for all $x \in \mathbb{R}^d$ and for every $k = 1, \dots, L$,

$$\|x\|_2 \, e^{-2c_A} \leq \left\|h_k^{x,(L)}\right\|_2 \leq \|x\|_2 \, e^{1.1c_A} \qquad \textit{and} \qquad \left\|M_k^{x,(L)} e_m\right\|_2 \leq e^{c_A}.$$

Note that we did not try to optimize the constants in front of the bounds, and one can easily sharpen them if needed.

Proof. We follow the same lines as [4]. Fix $L \geq 5c_A$. In the proof, we omit the explicit dependence in L. First, note that we can write the logarithm of the norm of the hidden state as follows:

$$\begin{aligned}\log\|h_k\| &= \log\|x\| + \frac{1}{2}\sum_{j=1}^{k} \log\frac{\|h_j\|^2}{\|h_{j-1}\|^2} \\ &= \log\|x\| + \frac{1}{2}\sum_{j=1}^{k} \log\left(1 + \underbrace{\frac{2\delta_L}{\|h_{j-1}\|^2}\Big\langle h_{j-1}, \sigma(\alpha_j h_{j-1})\Big\rangle + \delta_L^2 \frac{\|\sigma(\alpha_j h_{j-1})\|^2}{\|h_{j-1}\|^2}}_{=:\,\Delta_j}\right).\end{aligned}$$

We can bound Δ_j further:

$$\begin{aligned}\Delta_j &\leq 2\delta_L \|\alpha_j\|_F + \delta_L^2 \|\alpha_j\|_F^2 \\ &\leq 2c_A L^{-1} + c_A^2 L^{-2} \leq \frac{11}{5} c_A L^{-1}. \end{aligned} \tag{B.3}$$

The first inequality holds by Cauchy-Schwartz and Assumption 4.1 *(ii)*, the second by hypothesis, and the third by Assumption 4.1 *(i)*. Thus, we conclude the proof of the upper bound by noting that $\log(1+z) \leq z$ for all $z > -1$.

For the lower bound, first observe that Cauchy-Schwartz yields

$$\Delta_j \geq -2\delta_L \|\alpha_j\|_F \geq -2c_A L^{-1}.$$

From (B.3), we also have $|\Delta_j| \leq \frac{11}{25} < \frac{1}{2}$, so we can use the fact that $\log(1+z) \geq z - z^2$ for all $|z| < \frac{1}{2}$ to deduce that

$$\begin{aligned}\log \|h_k\| &\geq \log \|x\| + \frac{1}{2}\sum_{j=1}^{k}\left(\Delta_j - \Delta_j^2\right) \\ &\geq \log \|x\| - c_A - \frac{121}{25}c_A^2 L^{-1} \geq \log \|x\| - 2c_A,\end{aligned}$$

which concludes the proof for the lower bound on the hidden states.
For the upper bound on the Jacobians, we apply Lemma D.1 repeatedly on M_k to get

$$\begin{aligned}\log \|M_k e_m\|_2 &\leq \log \|e_m\|_2 + \sum_{j=k+1}^{L} \log \|I_d + \delta_L \mathrm{diag}\left(\nabla\sigma_d\left(\alpha_j h_{j-1}\right)\right)\alpha_j\|_2 \\ &\leq \sum_{j=k+1}^{L} \delta_L \|\mathrm{diag}\left(\nabla\sigma_d\left(\alpha_j h_{j-1}\right)\right)\alpha_j\|_2 \leq \sum_{j=k+1}^{L} \delta_L \|\alpha_j\|_F \leq c_A,\end{aligned}$$

where we use $\|\cdot\|_2 \leq \|\cdot\|_F$ and Assumption 4.1 *(ii)* in the third inequality. □

We deduce directly an upper bound on the loss function J_L that does not depend on L.

Corollary B.2. *Under the same hypotheses as Lemma B.1, we have*

$$J_L\left(\alpha^{(L)}\right) \leq 1 + e^{2.2c_A}.$$

Proof. By definition of the loss function and using Lemma B.1, we have

$$\begin{aligned}J_L\left(\alpha^{(L)}\right) &= \frac{1}{2N}\sum_{i=1}^{N}\left\|y_i - \widehat{y}_L\left(x_i, \alpha^{(L)}\right)\right\|_2^2 \\ &\leq \frac{1}{2N}\sum_{i=1}^{N} 2\|y_i\|^2 + 2\left\|h_L^{x_i,(L)}\right\|_2^2 \leq 1 + e^{2.2c_A}.\end{aligned}$$

□

B.3 Upper bounds on the gradient and Hessian of the loss function

Lemma B.3. *Let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ and $c_A > 0$ such that $L \geq 5c_A$ and $\left\|\alpha^{(L)}\right\|_{F,\infty} \leq c_A L^{-1/2}$. Then, under Assumption 4.1 (i)–(ii), for $k = 1, \ldots L$, it holds that*

$$\left\|\nabla_{\alpha_k} J_L\left(\alpha^{(L)}\right)\right\|_F^2 \leq 2de^{4.2c_A}L^{-1}J_L\left(\alpha^{(L)}\right).$$

Proof. Fix $L \geq 5c_A$. In the proof, we omit the explicit dependence in L. We first use Cauchy-Schwartz and (B.2) to bound the Frobenius norm.

$$\begin{aligned}
\|\nabla_{\alpha_k} J_L(\alpha)\|_F^2 &= \sum_{m,n=1}^{d} \left(\frac{\partial J_L}{\partial \alpha_{k,mn}}(\alpha) \right)^2 \\
&\leq \sum_{m,n=1}^{d} \frac{1}{N} \sum_{i=1}^{N} \left(\frac{\partial \ell}{\partial \alpha_{k,mn}} \left(y_i, \widehat{y}(x_i,\alpha)\right) \right)^2 \\
&\leq \sum_{m,n=1}^{d} \frac{\delta_L^2}{N} \sum_{i=1}^{N} \left(h_{k-1,n}^{x_i}\right)^2 \left\|\nabla_{\widehat{y}}\, \ell\left(y_i, \widehat{y}(x_i,\alpha)\right)\right\|_2^2 \left\|M_k^{x_i} e_m\right\|_2^2 \\
&= \frac{2L^{-1}}{N} \sum_{i=1}^{N} \left\|h_{k-1}^{x_i}\right\|_2^2 \ell(y_i, \widehat{y}(x_i,\alpha)) \left\|M_k^{x_i}\right\|_F^2 \\
&\leq 2de^{4.2c_A} L^{-1} J_L(\alpha),
\end{aligned}$$

where we use the fact that $2\ell(y,\widehat{y}) = \|\nabla_{\widehat{y}}\, \ell\,(y,\widehat{y})\|_2^2$ and Lemma B.1 in the last inequality. □

Finally, we derive an upper bound on the spectral norm of the Hessian of the loss function.

Lemma B.4. *Let $\alpha^{(L)} \in \mathbb{R}^{L\times d\times d}$ and $c_A > 0$ such that $L \geq 5c_A$ and $\left\|\alpha^{(L)}\right\|_{F,\infty} \leq c_A L^{-1/2}$. Then, under Assumption 4.1 (i)–(ii), we have*

$$\left\|\nabla_\alpha^2 J_L\left(\alpha^{(L)}\right)\right\|_2 \leq 5de^{4.3c_A}.$$

Proof. Fix $L \geq 5c_A$. In the proof, we omit the explicit dependence in L. We use first-order information (B.2) to compute the second-order derivatives. Straightforward but lengthy computations show that

$$\nabla_\alpha^2 J_L\left(\alpha^{(L)}\right) = H_{\text{psd}} + H + \widetilde{H} + \mathcal{O}\left(L^{-1/2}\right),$$

where $H_{\text{psd}}, H, \widetilde{H} \in \mathbb{R}^{Ld^2 \times Ld^2}$ are given by the following formulae:

$$H_{\text{psd}} = \frac{1}{N}\sum_{i=1}^{N} H_{\text{psd}}^{x_i} \quad H = \frac{1}{N}\sum_{i=1}^{N} H^{x_i} \quad \widetilde{H} = \frac{1}{N}\sum_{i=1}^{N} \widetilde{H}^{x_i},$$

where

$$\begin{aligned}
\left(H_{\text{psd}}^{x_i}\right)_{(k,\,mn)\,(k',\,m'n')} &= \delta_L^2 h_{k-1,n}^{x_i} h_{k'-1,n'}^{x_i} \dot{\sigma}_{k,x_i,m} \dot{\sigma}_{k',x_i,m'} \left(M_k^{x_i} e_m\right)^\top \left(M_{k'}^{x_i} e_{m'}\right) \\
H_{(k,\,mn)\,(k',\,m'n')}^{x_i} &= \delta_L h_{k-1,n}^{x_i} h_{k-1,n'}^{x_i} \ddot{\sigma}_{k,x_i,m} \left(\widehat{y}(x_i,\alpha) - y_i\right)^\top M_k^{x_i} e_m \mathbb{1}_{m=m'} \mathbb{1}_{k=k'} \\
\widetilde{H}_{(k,\,mn)\,(k',\,m'n')}^{x_i} &= \delta_L^2 h_{k-1,n}^{x_i} \dot{\sigma}_{k,x_i,m} \dot{\sigma}_{k,x_i,m'} \left(\widehat{y}(x_i,\alpha) - y_i\right)^\top M_{k,-k'}^{x_i} e_m \mathbb{1}_{m'=n'} \mathbb{1}_{k<k'}.
\end{aligned}$$

Here, $M_{k,-k'}^{x_i}$ is defined as the same product of matrices as $M_k^{x_i}$ in (B.1), but without the term $j = k'$. By the same reasoning as in Lemma B.1, we still have $\left\|M_{k,-k'}^{x_i} e_m\right\|_2 \leq e^{c_A}$. We readily see that for each i there exists Q_i such that $H_{\text{psd}}^{x_i} = Q_i^\top Q_i$, so H_{psd} is positive semi-definite. The trace of H_{psd} is straightforward to compute.

$$\begin{aligned}\operatorname{tr}\left(H_{\text{psd}}\right) = \frac{1}{N}\sum_{i=1}^{N} \operatorname{tr}\left(H_{\text{psd}}^{x_i}\right) &= \frac{\delta_L^2}{N}\sum_{i=1}^{N}\sum_{k,m,n} \left|h_{k-1,n}^{x_i}\right|^2 \left(\dot{\sigma}_{k,x_i,m}\right)^2 \left\|M_k^{x_i} e_m\right\|_2^2 . \\ &= \frac{1}{N}\sum_{i=1}^{N} L^{-1} \sum_{k=1}^{L} \left\|h_{k-1}^{x_i}\right\|_2^2 \left\|M_k^{x_i}\right\|_F^2 .\end{aligned}$$

We deduce that by Lemma B.1 that $\operatorname{tr}\left(H_{\text{psd}}\right) \leq d^2 e^{4.2 c_A}$.
The upper bound on the Frobenius norm of H and $\widetilde{H}$ is no harder.

$$\begin{aligned}\left\|H^{x_i}\right\|_F^2 &\leq \sum_{k=1}^{L}\sum_{m=1}^{d} \delta_L^2 \left\|h_{k-1}^{x_i}\right\|_2^4 \ell(y_i, \widehat{y}(x_i,\alpha)) \left\|M_k^{x_i} e_m\right\|_2^2 \leq d e^{6.4 c_A} \ell(y_i, \widehat{y}(x_i,\alpha)), \\ \left\|\widetilde{H}^{x_i}\right\|_F^2 &\leq \sum_{k \neq k'} \delta_L^4 \left\|h_{k-1}^{x_i}\right\|_2^2 \ell(y_i, \widehat{y}(x_i,\alpha)) d^2 e^{2 c_A} \leq d^2 e^{4.2 c_A} \ell(y_i, \widehat{y}(x_i,\alpha)).\end{aligned}$$

Hence, $\|H\|_F \leq \sqrt{2d} e^{3.2 c_A} J_L(\alpha)^{1/2}$ and $\|\widetilde{H}\|_F \leq \sqrt{2} d e^{2.1 c_A} J_L(\alpha)^{1/2}$. Using Corollary B.2 and wrapping both terms together, we get

$$\begin{aligned}\left\|\nabla_\alpha^2 J\left(\alpha^{(L)}\right)\right\|_2 &\leq \left\|H_{\text{psd}}\right\|_2 + \|H\|_2 + \|\widetilde{H}\|_2 + \mathcal{O}(L^{-1/2}) \\ &\leq \operatorname{tr}(H_{\text{psd}})^{1/2} + \|H\|_F + \|\widetilde{H}\|_F + \mathcal{O}(L^{-1/2}) \\ &\leq 5 d e^{4.3 c_A}.\end{aligned}$$

□

B.4 Lower bounds on loss gradients

This section contains a supporting result for the proof of Lemma 4.3.

Lemma B.5. *Let $\alpha^{(L)} \in \mathbb{R}^{L \times d \times d}$. Under Assumption 4.1 (i)–(ii), we have, for $k = 1, \ldots, L-1$,*

$$\begin{aligned}\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}} = &\frac{\delta_L}{N}\sum_{i=1}^{N} h_{k-1,n}^{x_i}\left(\dot{\sigma}_{k,x_i,m} - \dot{\sigma}_{k+1,x_i,m}\right) \nabla_{\widehat{y}}\, \ell\left(y_i, \widehat{y}(x_i,\alpha)\right)^\top M_{k+1}^{x_i} e_m \\ &+ \frac{\delta_L^2}{N}\sum_{i=1}^{N} \nabla_{\widehat{y}}\, \ell\left(y_i, \widehat{y}(x_i,\alpha)\right)^\top M_{k+1}^{x_i} \xi_{k,mn}^{x_i,(L)},\end{aligned}$$

where $\xi_{k,mn}^{x,(L)} \in \mathbb{R}^d$ *satisfies*

$$\left\|\xi_{k,mn}^{x,(L)}\right\|_2^2 \leq 2\left(h_{k-1,n}^x\right)^2 \|\alpha_{k+1}-\alpha_k\|_F^2 + 2\|\alpha_{k,n}\|_2^4 \left\|h_{k-1}^x\right\|_2^4.$$

Proof. We use the gradient computation (B.2) and the definition (B.1) to get

$$\frac{\partial J_L}{\partial \alpha_{k,mn}}\left(\alpha^{(L)}\right) = \frac{\delta_L}{N}\sum_{i=1}^N h_{k-1,n}^{x_i}\dot{\sigma}_{k,x_i,m}\nabla_{\widehat{y}}\,\ell\left(y_i, \widehat{y}(x_i,\alpha^{(L)})\right)^\top M_k^{x_i} e_m,$$

$$\frac{\partial J_L}{\partial \alpha_{k+1,mn}}\left(\alpha^{(L)}\right) = \frac{\delta_L}{N}\sum_{i=1}^N \left(h_{k-1,n}^{x_i} + \delta_L \sigma_{k,x_i,n}\right)\dot{\sigma}_{k+1,x_i,m}\nabla_{\widehat{y}}\,\ell\left(y_i, \widehat{y}(x_i,\alpha^{(L)})\right)^\top M_{k+1}^{x_i} e_m.$$

We use the identity $M_k^{x_i} = M_{k+1}^{x_i}\Big(I_d + \delta_L \mathrm{diag}\left(\dot{\sigma}_{k+1,x_i}\right)\alpha_{k+1}\Big)$ and we take the difference of the two equations above to get

$$\begin{aligned}\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}} &= \frac{\delta_L}{N}\sum_{i=1}^N h_{k-1,n}^{x_i}\left(\dot{\sigma}_{k,x_i,m} - \dot{\sigma}_{k+1,x_i,m}\right)\nabla_{\widehat{y}}\,\ell\left(y_i,\widehat{y}(x_i,\alpha)\right)^\top M_{k+1}^{x_i} e_m \\ &\quad + \frac{\delta_L^2}{N}\sum_{i=1}^N \nabla_{\widehat{y}}\,\ell\left(y_i,\widehat{y}(x_i,\alpha)\right)^\top M_{k+1}^{x_i}\xi_{k,mn}^{x_i,(L)},\end{aligned}$$

where

$$\begin{aligned}\left\|\xi_{k,mn}^{x,(L)}\right\|_2^2 &\leq 2\left(h_{k-1,n}^x\right)^2\|\alpha_{k+1}-\alpha_k\|_F^2 + 2\left(\dot{\sigma}_{k+1,x,m}\right)^2\left(\sigma\left(\alpha_k h_{k-1}^x\right)_n - \left(\alpha_k h_{k-1}^x\right)_n\right)^2 \\ &\leq 2\left(h_{k-1,n}^x\right)^2\|\alpha_{k+1}-\alpha_k\|_F^2 + 2\|\alpha_{k,n}\|_2^4\left\|h_{k-1}^x\right\|_2^4.\end{aligned}$$

We use the fact that $|\sigma(z) - z| \leq z^2$ by Assumption 4.1 *(i)*. □

B.5 Weight norms and loss function under gradient descent

This section contains the proof of Lemmas 4.4 and 4.5.

Proof of Lemma 4.4 Fix $L \in \mathbb{N}^*$. In the proof, we omit the explicit dependence in L. We use the identity $\frac{1}{2}(A^2 - B^2) = B(A-B) + \frac{1}{2}(A-B)^2$ and the gradient descent update rule to first compute

$$f_{k,m}(\widetilde{\alpha}) - f_{k,m}(\alpha) = \underbrace{-L\eta_L\sum_{n=1}^d \alpha_{k,mn}\frac{\partial J_L}{\partial \alpha_{k,mn}}(\alpha)}_{=:S_1(\alpha)} + \underbrace{\frac{L\eta_L^2}{2}\sum_{n=1}^d\left(\frac{\partial J_L}{\partial \alpha_{k,mn}}\right)^2(\alpha)}_{=:S_2(\alpha)}. \qquad \text{(B.4)}$$

Recall that the gradient of the loss ℓ with respect to the parameter $\alpha_{k,mn}$ at sample (x, y) is given by (B.2), so that we can compute

$$\frac{\partial J_L}{\partial \alpha_{k,mn}}(\alpha) = \frac{\delta_L}{N} \sum_{i=1}^{N} h_{k-1,n}^{x_i}(\alpha)\, \dot{\sigma}_{k,x_i,m}(\alpha)\, \nabla_{\widehat{y}}\, \ell\left(y_i, \widehat{y}\left(x_i, \alpha\right)\right)^\top M_k^{x_i}(\alpha)\, e_m.$$

Recall also from (4.9) that

$$G_k^{x,y}(\alpha) \cdot e_m = \frac{\partial \ell(y, \cdot)}{\partial h_k}\left(\widehat{y}(x, \alpha)\right) e_m = \nabla_{\widehat{y}}\, \ell\left(y, \widehat{y}(x, \alpha)\right)^\top M_k^x(\alpha)\, e_m.$$

We focus on the square of first order term $S_{1,m}(\alpha)$ defined above. We have

$$\begin{aligned}
S_1(\alpha)^2 &= \frac{L^2 \delta_L^2 \eta_L^2}{N^2} \left(\sum_{n=1}^{d} \alpha_{k,mn} \sum_{i=1}^{N} h_{k-1,n}^{x_i}(\alpha)\, \dot{\sigma}_{k,x_i,m}(\alpha)\, G_k^{x_i,y_i}(\alpha) \cdot e_m \right)^2 \\
&\leq L^2 \delta_L^2 \eta_L^2 \sum_{n=1}^{d} \alpha_{k,mn}^2 \sum_{n=1}^{d} \left(\frac{1}{N} \sum_{i=1}^{N} h_{k-1,n}^{x_i}(\alpha)\, \dot{\sigma}_{k,x_i,m}(\alpha)\, G_k^{x_i,y_i}(\alpha) \cdot e_m \right)^2 \\
&\leq 2L \delta_L^2 \eta_L^2 f_{k,m}(\alpha) \sum_{n=1}^{d} \frac{1}{N} \sum_{i=1}^{N} \left(h_{k-1,n}^{x_i}(\alpha)\, \dot{\sigma}_{k,x_i,m}(\alpha)\, G_k^{x_i,y_i}(\alpha) \cdot e_m \right)^2 \\
&\leq 2\eta_L^2 f_{k,m}(\alpha) \frac{1}{N} \sum_{i=1}^{N} \left\| h_{k-1}^{x_i}(\alpha) \right\|_2^2 \left\| G_k^{x_i,y_i}(\alpha) \right\|_\infty^2 .
\end{aligned}$$

We used twice the Cauchy-Schwarz inequality and Assumption 4.1 *(i)-(ii)*. Define now

$$r_k(\alpha) \coloneqq \eta_L \left(\frac{1}{N} \sum_{i=1}^{N} \left\| h_{k-1}^{x_i}(\alpha) \right\|_2^2 \left\| G_k^{x_i,y_i}(\alpha) \right\|_\infty^2 \right)^{1/2} .$$

By similar estimations, we also upper bound the second-order term: $S_2(\alpha) \leq \frac{1}{2} r_k(\alpha)^2$. Equation (B.4) then yields to

$$\begin{aligned}
f_{k,m}(\widetilde{\alpha}) &\leq f_{k,m}(\alpha) + |S_1(\alpha)| + S_2(\alpha) \\
&\leq f_{k,m}(\alpha) + \sqrt{2} r_k(\alpha) f_{k,m}(\alpha)^{1/2} + \frac{1}{2} r_k(\alpha)^2 = \left(f_{k,m}(\alpha)^{1/2} + \frac{1}{\sqrt{2}} r_k(\alpha) \right)^2 .
\end{aligned}$$

□

Proof of Lemma 4.5 Fix $L \in \mathbb{N}^*$. In the proof, we omit the explicit dependence in L. Define $g_{k,m}(\alpha) \coloneqq \frac{1}{2} L^2 \left\| \alpha_{k+1,m} - \alpha_{k,m} \right\|_2^2$ so that $g_k = \sum_{m=1}^{d} g_{k,m}$. We also omit the dependence in α when it is clear. We use the identity $\frac{1}{2}(A^2 - B^2) = B(A-B) + \frac{1}{2}(A-B)^2$

and the gradient descent update rule to first compute

$$g_{k,m}(\widetilde{\alpha}) - g_{k,m}(\alpha) = \underbrace{L^2\eta_L \sum_{n=1}^{d} (\alpha_{k+1,mn} - \alpha_{k,mn}) \left(\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}} \right)(\alpha)}_{=: S_{1,m}(\alpha)}$$
$$+ \underbrace{\frac{L^2\eta_L^2}{2} \sum_{n=1}^{d} \left(\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}} \right)^2 (\alpha)}_{=: S_{2,m}(\alpha)} .$$

Next, we use Lemma B.5 to estimate the difference of gradients with respect to weights in neighbouring layers. We also use the fact that $L \geq 5c_A$ and $\left\| \alpha^{(L)} \right\|_{F,\infty} \leq c_A L^{-1/2}$ to apply Lemma B.1. Recall the definition of G in (4.9).

$$\frac{\partial J_L}{\partial \alpha_{k,mn}} - \frac{\partial J_L}{\partial \alpha_{k+1,mn}} = \frac{\delta_L}{N} \sum_{i=1}^{N} h^{x_i}_{k-1,n} \left(\dot{\sigma}_{k,x_i,m} - \dot{\sigma}_{k+1,x_i,m} \right) G^{x_i,y_i}_{k+1} \cdot e_m$$
$$+ \frac{\delta_L^2}{N} \sum_{i=1}^{N} \nabla_{\widehat{y}} \ell \left(y_i, \widehat{y}(x_i,\alpha) \right)^\top M^{x_i}_{k+1} \xi^{x_i,\,(L)}_{k,mn},$$

where $\xi^{x,\,(L)}_{k,mn} \in \mathbb{R}^d$ satisfies

$$\sum_{n=1}^{d} \left\| \xi^{x,\,(L)}_{k,mn} \right\|_2^2 \leq 4c_A^2 e^{2.2c_A} L^{-1}. \tag{B.5}$$

We focus on the first order term $S_{1,m}(\alpha)$ defined above. We have

$$S_{1,m}(\alpha) = \frac{\eta_L \delta_L L^2}{N} \sum_{i=1}^{N} G^{x_i,y_i}_{k+1} \cdot e_m \left(\dot{\sigma}_{k,x_i,m} - \dot{\sigma}_{k+1,x_i,m} \right) \sum_{n=1}^{d} \left(\alpha_{k+1,mn} - \alpha_{k,mn} \right) h^{x_i}_{k-1,n}$$
$$+ \frac{\eta_L \delta_L^2 L^2}{N} \sum_{i=1}^{N} \nabla_{\widehat{y}} \ell \left(y_i, \widehat{y}(x_i,\alpha) \right)^\top M^{x_i}_{k+1} \sum_{n=1}^{d} \left(\alpha_{k+1,mn} - \alpha_{k,mn} \right) \xi^{x_i}_{k,mn}.$$

Now, as σ' is 1−Lipschitz, we can write

$$|S_{1,m}(\alpha)| \leq \frac{\eta_L \delta_L L^2}{N} \sum_{i=1}^{N} \left\| G^{x_i,y_i}_{k+1,} \right\|_\infty \left| \left(\alpha_{k+1} - \alpha_k \right) h^{x_i}_{k-1} \right|_m \left| \alpha_k h_{k-1} - \alpha_{k+1} h_k \right|_m$$
$$+ \eta_L \delta_L^2 L^2 \left[\frac{2}{N} \sum_{i=1}^{N} \ell \left(y_i, \widehat{y}(x_i,\alpha) \right) \left\| M^{x_i}_{k+1} \right\|_2^2 \left\| \alpha_{k+1,m} - \alpha_{k,m} \right\|_2^2 \sum_{n=1}^{d} \left\| \xi^{x_i}_{k,mn} \right\|_2^2 \right]^{1/2}.$$

We now use the fact that $L \geq 5c_A$ and $\left\|\alpha^{(L)}\right\|_{F,\infty} \leq c_A L^{-1/2}$ to apply Lemma B.1 on the second term and deduce that

$$\begin{aligned}
|S_{1,m}(\alpha)| \leq{}& \frac{\eta_L \delta_L L^2}{N} \sum_{i=1}^{N} \left\|G_{k+1}^{x_i,y_i}\right\|_\infty \left|(\alpha_{k+1} - \alpha_k)\, h_{k-1}^{x_i}\right|_m^2 \\
&+ \frac{\eta_L \delta_L L^2}{N} \sum_{i=1}^{N} \left\|G_{k+1}^{x_i,y_i}\right\|_\infty \left|(\alpha_{k+1} - \alpha_k)\, h_{k-1}^{x_i}\right|_m \left|\alpha_{k+1}\left(h_k^{x_i} - h_{k-1}^{x_i}\right)\right|_m \\
&+ 2e^{c_A}\eta_L \left(\sum_{n=1}^{d} \left\|\xi_{k,mn}^{x_i}\right\|_2^2\right)^{1/2} g_{k,m}(\alpha)^{1/2} J_L(\alpha)^{1/2}.
\end{aligned}$$

We apply Cauchy-Schwarz to the first and second term and equation (B.5) to the third term to get

$$\begin{aligned}
|S_{1,m}(\alpha)| \leq{}& 2\eta_L L^{-1/2} \frac{1}{N} \sum_{i=1}^{N} \left\|h_{k-1}^{x_i}\right\|_2^2 \left\|G_{k+1}^{x_i,y_i}\right\|_\infty g_{k,m}(\alpha) \\
&+ \eta_L \delta_L L^2 \left[\frac{1}{N} \sum_{i=1}^{N} \left\|G_{k+1}^{x_i,y_i}\right\|_\infty^2 \left\|\alpha_{k+1,m} - \alpha_{k,m}\right\|_2^2 \left\|h_{k-1}^{x_i}\right\|_2^2 \left\|\alpha_{k+1,m}\right\|_2^2 \delta_L^2 \left\|\sigma_{k,x_i}\right\|_2^2\right]^{1/2} \\
&+ 4c_A e^{2.1c_A} \eta_L L^{-1/2} g_{k,m}(\alpha)^{1/2} J_L(\alpha)^{1/2}.
\end{aligned}$$

We now use Lemma B.1 and the identity $G_{k+1}^{x_i,y_i} = M_{k+1}^{x_i}\left(\widehat{y}(x_i,\alpha) - y_i\right)$ to estimate the second term in the RHS:

$$\begin{aligned}
|S_{1,m}(\alpha)| \leq{}& 2\eta_L L^{-1/2} \frac{1}{N} \sum_{i=1}^{N} \left\|h_{k-1}^{x_i}\right\|_2^2 \left\|G_{k+1}^{x_i,y_i}\right\|_\infty g_{k,m}(\alpha) \\
&+ 2c_A^2 e^{3.2c_A} \eta_L L^{-1} g_{k,m}(\alpha)^{1/2} J_L(\alpha)^{1/2} + 4c_A e^{2.1c_A} \eta_L L^{-1/2} g_{k,m}(\alpha)^{1/2} J_L(\alpha)^{1/2}.
\end{aligned}$$

Thus,

$$\begin{aligned}
|S_{1,m}(\alpha)| \leq{}& 2\eta_L L^{-1/2} \frac{1}{N} \sum_{i=1}^{N} \left\|h_{k-1}^{x_i}\right\|_2^2 \left\|G_{k+1}^{x_i,y_i}\right\|_\infty g_{k,m}(\alpha) \\
&+ 2c_A e^{2.1c_A} \eta_L \left(c_A e^{1.1c_A} L^{-1} + 2L^{-1/2}\right) g_{k,m}(\alpha)^{1/2} J_L(\alpha)^{1/2}.
\end{aligned}$$

Define

$$\begin{aligned}
r_k(\alpha) &\coloneqq \eta_L \frac{1}{N} \sum_{i=1}^{N} \left\|h_{k-1}^{x_i}\right\|_2^2 \left\|G_{k+1}^{x_i,y_i}\right\|_\infty \\
\mathcal{E}_{k,m}(L,d,\alpha) &\coloneqq c_A e^{2.1c_A} \eta_L \left(c_A e^{1.1c_A} L^{-1} + 2L^{-1/2}\right) g_{k,m}(\alpha)^{1/2} J_L(\alpha)^{1/2}.
\end{aligned}$$

We then have $|S_{1,m}(\alpha)| \leq 2L^{-1/2} g_{k,m}(\alpha) r_k(\alpha) + 2\mathcal{E}_{k,m}(L,d,\alpha)$. We use similar techniques to derive the upper bound $S_{2,m}(\alpha) \leq L^{-1} g_{k,m}(\alpha) r_k(\alpha)^2 + \mathcal{O}(\mathcal{E}_{k,m}(L,d,\alpha))$.

Hence, we deduce the following recurrence relation.

$$g_{k,m}(\widetilde{\alpha}) \leq g_{k,m}(\alpha)+|S_{1,m}(\alpha)|+S_{2,m}(\alpha) \leq g_{k,m}(\alpha)\left(1+L^{-1/2}r_k(\alpha)\right)^2+\mathcal{O}\left(\mathcal{E}_{k,m}(L,d,\alpha)\right).$$

Summing over $m = 1, \ldots, d$ and using Cauchy-Schwarz on the $\mathcal{E}_{k,m}$ terms, we get

$$g_k(\widetilde{\alpha}) \leq g_k(\alpha)\left(1+L^{-1/2}r_k(\alpha)\right)^2+\mathcal{O}\left(\mathcal{E}_k(L,d,\alpha)\right),$$

where

$$\mathcal{E}_k(L,d,\alpha) \coloneqq c_A e^{2.1c_A}\eta_L\left(c_A e^{1.1c_A}L^{-1}+2L^{-1/2}\right)g_k(\alpha)^{1/2}J_L(\alpha)^{1/2}. \tag{B.6}$$

□

B.6 Supporting lemma for Theorem 4.6

Lemma B.6. *Let $\alpha^{(L)}(0) \in \mathbb{R}^{L\times d\times d}$ be any weight initialization. Define recursively $\alpha^{(L)}(t+1) = \alpha^{(L)}(t) - \eta_L(t)\nabla_\alpha J_L\big(\alpha^{(L)}(t)\big)$ for $t = 0, \ldots, T-1$. Assume that for all $t = 0, \ldots, T-1$, there exist $c_A(t), \underline{c}(t), \bar{c}(t) > 0$ such that*

(i) $L \geq 5\max_{t<T} c_A(t)$,

(ii) $\left\|\alpha^{(L)}(t)\right\|_{F,\infty} \leq c_A(t)L^{-1/2}$, *and*

(iii) $\left\|\nabla_{\alpha^{(L)}} J_L\big(\alpha^{(L)}(t)\big)\right\|_F^2 \geq \underline{c}(t)J_L\big(\alpha^{(L)}(t)\big) - \bar{c}(t)L^{-1}$.

Then, under Assumption 4.1 (i)–(ii), if the learning rates satisfy:

$$\eta_L(t) < \min\left(\frac{1}{2}c_A(t)e^{-3.2c_A(t)}, \frac{1}{10}\underline{c}(t)d^{-1}e^{-8.5c_A(t)}\right),$$

we have, for each $t = 0, \ldots, T$:

$$J_L\big(\alpha^{(L)}(t)\big) \leq \exp\left(-\frac{1}{2}\sum_{t'=0}^{t-1}\underline{c}(t')\eta_L(t')\right)J_L\big(\alpha^{(L)}(0)\big) + L^{-1}\sum_{t'=0}^{t-1}\bar{c}(t')\eta_L(t'). \tag{B.7}$$

Proof. Fix $L \geq 5\max_{t<T} c_A(t)$. We omit the explicit dependence in L. Fix $t \in [0, T)$. We first view $\alpha(t)$, $\nabla_\alpha J_L(\alpha(t)) \in \mathbb{R}^{L\times d\times d}$ as vectors in the Euclidean space $\mathbb{R}^{Ld^2}$, and we get by hypothesis and by Lemma B.3 that

$$\|\mathrm{vec}(\alpha(t))\|_2 = \|\alpha(t)\|_F = \left(\sum_{k=1}^{L}\|\alpha_k(t)\|_F^2\right)^{1/2} \leq c_A(t),$$
$$\underline{c}(t)J_L(\alpha(t)) - \bar{c}(t)L^{-1} = \|\nabla_\alpha J_L(\alpha(t))\|_F^2 \leq 2e^{4.2c_A(t)}J_L(\alpha(t)).$$

We want to use Lemma D.3 with $p = Ld^2$, $R = c_A$, $x_0 = \alpha(t)$ and $x = \alpha(t) - \eta_L(t)\nabla_\alpha J_L(\alpha(t))$. For this, we need to check two assumptions. The first is an upper bound on the spectral norm of the Hessian of J_L, which we get from Lemma B.4.

$$H_\infty(t) = \sup_{\|\alpha'\|_F \leq c_A(t)} \left\|\nabla^2 J_L(\alpha')\right\|_2 \leq 5de^{4.3c_A(t)}.$$

The second is an upper bound on the norm of $x - x_0 = -\eta_L(t)\nabla_\alpha J_L(\alpha(t))$, which we get from Lemma B.3.

$$\begin{aligned}\eta_L(t) \left\|\nabla_\alpha J_L(\alpha(t))\right\|_2 &\leq \sqrt{2}\eta_L(t)e^{2.1c_A(t)} J_L(\alpha(t))^{1/2} \\ &\leq \sqrt{2}\eta_L(t)e^{2.1c_A(t)} \left(1 + e^{2.2c_A(t)}\right)^{1/2} \\ &\leq 2\eta_L(t)e^{3.2c_A(t)} \leq c_A(t),\end{aligned}$$

where the second inequality comes from Corollary B.2 and the third inequality from the fact that $(1+z)^{1/2} \leq (2z)^{1/2}$ for $z \geq 1$. Hence, we can apply Lemma D.3 and deduce that

$$\begin{aligned}J_L\left(\alpha(t+1)\right) &= J_L\Big(\alpha(t) - \eta_L(t)\nabla_\alpha J_L(\alpha(t))\Big) - J_L(\alpha(t)) \\ &\leq J_L\left(\alpha(t)\right) - \eta_L(t)\left\|\nabla_\alpha J_L(\alpha(t))\right\|_F^2 + \frac{1}{2}H_\infty(t)\eta_L(t)^2\left\|\nabla_\alpha J_L(\alpha(t))\right\|_2^2 \\ &\leq \left(1 - \underline{c}(t)\eta_L(t) + 5d\eta_L(t)^2 e^{8.5c_A(t)}\right) J_L\left(\alpha(t)\right) + \bar{c}(t)\eta_L(t)L^{-1}.\end{aligned}$$

To finish the proof, we apply Lemma D.4 *(i)* with

$$u_L(t) \coloneqq \underline{c}(t)\eta_L(t) - 5d\eta_L(t)^2 e^{8.5c_A(t)} \geq \frac{1}{2}\underline{c}(t)\eta_L(t) > 0,$$

and the fact that $1 - x \leq e^{-x}$. Hence,

$$\begin{aligned}J_L(\alpha(T)) &\leq \exp\left(-\sum_{t=0}^{T-1} u_L(t)\right) J_L(\alpha(0)) + L^{-1}\sum_{t=0}^{T-1}\bar{c}(t)\eta_L(t) \\ &\leq \exp\left(-\frac{1}{2}\sum_{t=0}^{T-1}\underline{c}(t)\eta_L(t)\right) J_L(\alpha(0)) + L^{-1}\sum_{t=0}^{T-1}\bar{c}(t)\eta_L(t).\end{aligned}$$

□

Appendix C

Technical results of Chapter 5

C.1 Properties of path-homogeneous functions

Lemma C.1. *Let* $F(\mu) = L\left(\int_\Theta \Phi(\theta)\mathrm{d}\mu(\theta)\right) + \int_\Theta V(\theta)\mathrm{d}\mu(\theta)$*. Assume that* Φ *and* V *are* (α, k)*-homogeneous. Then*

$$\min_{\mu \in \mathcal{M}_+(\Theta)} F(\mu) = \min_{\mu \in \mathcal{P}(\Theta)} F(\mu)$$

Proof. It suffices to show that for every non-negative measure $\mu \in \mathcal{M}_+(\Theta)$, there exists a corresponding probability measure $\nu \in \mathcal{P}(\Theta)$ that has the same functional value $F(\mu) = F(\nu)$.

Let $M := \mu(\Theta) > 0$ be the mass of the parameter set Θ and define a mapping $T\colon \Theta \to \Theta$ as $T(\theta) = M^{\alpha/k} \odot \theta$. We define $\nu := T_\#\left(\mu/M\right)$ to be the pushforward measure of μ by T. Then the following holds

$$\nu(\Theta) = \int_\Theta \mathrm{d}T_\#(\mu/M) = \frac{1}{M}\int_\Theta \mathrm{d}\mu = 1,$$

Moreover, as Φ is (α, k)-homogeneous,

$$\begin{aligned}
\int_\Theta \Phi(\theta)\mathrm{d}\mu(\theta) &= \frac{1}{M}\int_\Theta \left(M^{1/k}\right)^k \Phi(\theta)\mathrm{d}\mu(\theta) && \text{(C.1)}\\
&= \frac{1}{M}\int_\Theta \Phi\left(T(\theta)\right)\mathrm{d}\mu(\theta) && \text{(C.2)}\\
&= \int_\Theta \Phi(\theta)\mathrm{d}\nu(\theta), && \text{(C.3)}
\end{aligned}$$

where in the first line we divide and multiply by M, in the second line we use the (α, k)-homogeneity of Φ with $\lambda := M^{1/k}$, and in the last line we use the definition of a pushforward measure by T. The same analysis applies *mutatis mutandis* for V. □

We continue by describing geometric properties of path-homogeneous functions that are necessary to prove the convergence results. Let $f\colon \mathbb{R}^d \to U$ be an $(\alpha, k)-$homogeneous function for which we define the following ellipse

$$\mathcal{E} := \left\{x \in \mathbb{R}^d : \|x\|_A = 1\right\}, \tag{C.4}$$

where $\|\cdot\|_A$ denotes the Mahalanobis norm $\|x\|_A = \sqrt{x^T A x}$ and A is a diagonal matrix whose entries are $A_{i,i} = \alpha_i$.

For a fixed $x \in \mathbb{R}^d \setminus \{0\}$, we define the path $p_x\colon \mathbb{R}_{>0} \to \mathbb{R}^d$ by $p_x(\lambda) := \lambda^\alpha \odot x$ so that

$$f\left(p_x(\lambda)\right) = f(\lambda^\alpha \odot x) = \lambda^k f(x). \tag{C.5}$$

Along p_x, f becomes a monomial of degree k in λ, with scaling coefficient $f(x)$.

Define a projection onto the ellipse $\pi_{\mathcal{E}}\colon \mathbb{R}^d \setminus \{0\} \to \mathcal{E}$ along the curves $(p_x)_{x\in\mathcal{E}}$. The following lemma ensures that this is well-defined in the sense that for any $x \in \mathbb{R}^d \setminus \{0\}$, there is a unique point $\xi \in \mathcal{E}$ on the ellipse such that x is on $p_\xi\left(\mathbb{R}_{>0}\right) := \{p_\xi\left(\lambda\right) : \lambda > 0\}$.

Lemma C.2. *The paths $\{p_\xi : \xi \in \mathcal{E}\}$ form a disjoint union of the space*

$$\bigsqcup_{\xi\in\mathcal{E}} p_\xi\left(\mathbb{R}_{>0}\right) = \mathbb{R}^d \setminus \{0\} \tag{C.6}$$

and each path intersects the ellipse exactly at one point

$$\forall \xi \in \mathcal{E} : \quad \mathcal{E} \cap p_\xi\left(\mathbb{R}_{>0}\right) = \{\xi\}. \tag{C.7}$$

Proof. Let $\xi_1, \xi_2 \in \mathcal{E}$ with $\xi_1 \neq \xi_2$. We first show that $p_{\xi_1}\left(\mathbb{R}_{>0}\right) \cap p_{\xi_2}\left(\mathbb{R}_{>0}\right) = \emptyset$. Suppose that $x \in \mathbb{R}^d$ is in the intersection of the two paths, then there must exist $\lambda_1, \lambda_2 > 0$ such that $x = \lambda_1^\alpha \odot \xi_1 = \lambda_2^\alpha \odot \xi_2$.

Clearly, if $\lambda_1 = \lambda_2$ then $\xi_1 = \xi_2$ which is a contradiction. Therefore, consider the case where $\lambda_1 \neq \lambda_2$ and assume without loss of generality that $\lambda_1 > \lambda_2$. We then have the following relation $\xi_2 = \left(\lambda_1/\lambda_2\right)^\alpha \odot \xi_1$ and we deduce that

$$1 = \|\xi_2\|_A^2 = \sum_{i=1}^d \alpha_i (\xi_2)_i^2 = \sum_{i=1}^d \alpha_i \left(\frac{\lambda_1}{\lambda_2}\right)^{2\alpha_i} (\xi_1)_i^2 > \sum_{i=1}^d \alpha_i (\xi_1)_i^2 = \|\xi_1\|_A^2 = 1, \tag{C.8}$$

which results into a contradiction. Hence, the paths for $\xi_1 \neq \xi_2$ do not intersect, i.e. $p_{\xi_1}\left(\mathbb{R}_{>0}\right) \cap p_{\xi_2}\left(\mathbb{R}_{>0}\right) = \emptyset$.

We proceed to prove that the paths cover the whole of $\mathbb{R}^d \setminus \{0\}$. Let $x \in \mathbb{R}^d \setminus \{0\}$ be an arbitrary point for which we show that it lies on a path p_{ξ_0}. Define the following function $g : \mathbb{R}_{>0} \to \mathbb{R}$

$$g(\lambda) := \sum_{i=1}^d \alpha_i \lambda^{-2\alpha_i} x_i^2. \tag{C.9}$$

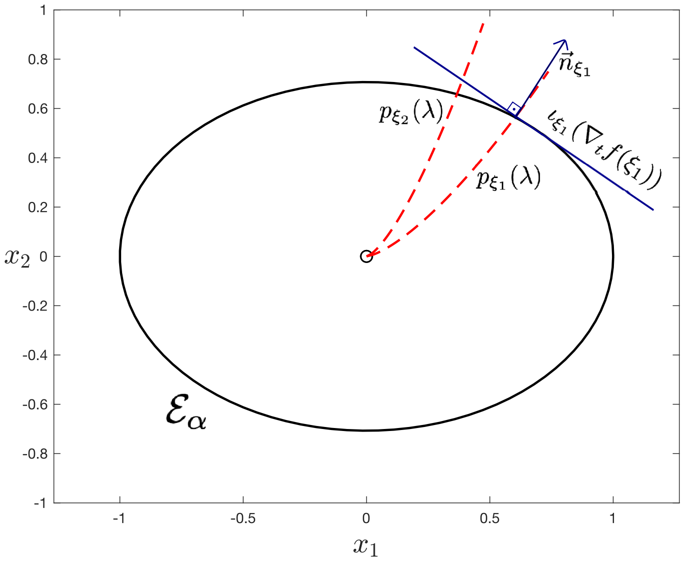

Figure C.1: Illustration of the case $\alpha = (1, 2)$. The ellipse is $\mathcal{E}_\alpha = \{x \in \mathbb{R}^2 : x_1^2 + 2x_2^2 = 1\}$. The paths $\lambda \in \mathbb{R}_{>0} \mapsto p_{\xi_i}(\lambda) \in \mathbb{R}^2$ are displayed in red for two distinct points $\xi_1, \xi_2 \in \mathcal{E}_\alpha$ on the ellipse. The outer normal unit vector of the ellipse $\mathcal{E}_\alpha$ at ξ_1 is denoted by $\vec{n}_{\xi_1}$. The tangential component of the gradient at ξ_1 of an $(\alpha, k)-$homogeneous f is denoted $\iota_{\xi_1}(\nabla_t f(\xi_1))$ and lies in the linear space spanned by the tangent line of the ellipse $\mathcal{E}_\alpha$ at ξ_1.

The function g is continuous and has two limits, $\lim_{\lambda\to+\infty} g(\lambda) = 0$ and $\lim_{\lambda\downarrow 0} g(\lambda) = +\infty$, therefore by the intermediate value theorem there must exist $\lambda_0 > 0$ such that $g(\lambda_0) = 1$. Let $\xi_0 = \lambda_0^{-\alpha} \odot x$ which by the definition of g must be a point on the ellipse $\xi_0 \in \mathcal{E}$ and $x = p_{\xi_0}(\lambda_0)$. This can be done for any $x \in \mathbb{R}^d \setminus \{0\}$ which proves that the paths cover the whole set $\mathbb{R}^d \setminus \{0\}$.

Finally, let $\xi \in \mathcal{E}$. We show that if there exists $\lambda > 0$ such that $p_\xi(\lambda) \in \mathcal{E}$, then $\lambda = 1$. To that end, consider the function $\tilde{g} : \mathbb{R}_{>0} \to \mathbb{R}$ defined by $\tilde{g}(\lambda) := \sum_{i=1}^d \alpha_i \lambda^{2\alpha_i} \xi_i^2$. We have that $p_\xi(\lambda) \in \mathcal{E} \iff \tilde{g}(\lambda) = 1$. But $\tilde{g}$ is an increasing function, so $\lambda = 1$ is the unique solution to $\tilde{g}(\lambda) = 1$. □

For $x \in \mathbb{R}^d \setminus \{0\}$, we can thus uniquely write $x = \lambda(x)^\alpha \odot \pi_\mathcal{E}(x)$. We prove that $\lambda(\cdot)$ grows at a sublinear rate.

Corollary C.3. *Let $\lambda \colon \mathbb{R}^d \setminus \{0\} \to \mathbb{R}_{>0}$ such that $x \mapsto \lambda(x)$ is defined implicitly by $x = p_{\pi_\mathcal{E}(x)}(\lambda(x))$, where $\pi_\mathcal{E}(x) \in \mathcal{E}$ is the unique point on $\mathcal{E}$ such that $x \in p_{\pi_\mathcal{E}(x)}(\mathbb{R}_{>0})$. Then λ is well-defined and continuous. Furthermore, we have*

$$\lambda(x) \leq \max(1, \|x\|_A), \quad \forall x \in \mathbb{R}^d \setminus \{0\}.$$

Proof. The first part is a direct consequence of the strict monotonicity of the function g defined in the proof of Lemma C.2, using the inverse function theorem. For the second part, observe that if $\lambda(x) > 1$, we have $\|x\|_A^2 = \|\lambda(x)^\alpha \odot \pi_\mathcal{E}(x)\|_A^2 \geq \|\lambda(x)^{\mathbb{1}_d} \odot \pi_\mathcal{E}(x)\|_A^2 = \lambda(x)^2$ as $\alpha_i \geq 1$, $\forall i = 1, \dots d$. □

We now have an adaptation of Euler's homogeneous function theorem [135] for path-homogeneous functions.

Remark C.4. *If a function $f \in C^1\left(\mathbb{R}^d \backslash \{0\}, \mathbb{R}\right)$ is $(\alpha, k)-$homogeneous, then $\nabla f(x)^T Ax = kf(x)$, for each $x \in \mathbb{R}^d \backslash \{0\}$ and $\lambda > 0$. Indeed, by differentiating the identity $f(\lambda^\alpha \odot x) = \lambda^k f(x)$ with respect to λ and evaluating at $\lambda = 1$ we get*

$$\nabla f(x)^\top Ax = \frac{\mathrm{d}}{\mathrm{d}\lambda} f(\lambda^\alpha \odot x)\big|_{\lambda=1} = \frac{\mathrm{d}}{\mathrm{d}\lambda} \lambda^k f(x)\big|_{\lambda=1} = kf(x).$$

As a consequence, the decomposition of the gradient of a (α, k)–homogenous function at $\xi \in \mathcal{E}$ can be decomposed explicitly into its normal component orthogonal to $\mathcal{E}$ and its tangential component belonging to $T_\xi\mathcal{E}$.

Remark C.5. *For $\xi \in \mathcal{E}$, we denote n_ξ the outer normal unit vector to $\mathcal{E}$. As $\mathcal{E}$ is a level set of $x \mapsto \|x\|_A$, we directly have $n_\xi = A\xi / \|A\xi\|_2$. Hence, if f is a (α, k)–homogenous function, we have*

$$\nabla f(\xi) = \nabla_n f(\xi) n_\xi + \iota_\xi(\nabla_t f\big|_\mathcal{E}(\xi)) = \frac{kf(\xi)}{\|A\xi\|_2} n_\xi + \iota_\xi(\nabla_t f\big|_\mathcal{E}(\xi)), \tag{C.10}$$

where we denote ∇_n the directional derivative along the normal vector n_ξ, ∇_t the gradient expressed in an orthonormal basis of $T_\xi\mathcal{E}$ and $\iota_\xi : T_\xi\mathcal{E} \hookrightarrow \operatorname{span}(n_\xi)^\perp \subset \mathbb{R}^d$. The second equality in (C.10) follows by Remark C.4.

C.2 Auxiliary results and proofs of Section 5.3

C.2.1 Bound on the variation of the subgradient

In the following section, let g_μ be the restriction of $F'(\mu)$ to the ellipse $\mathcal{E}$

$$g_\mu = F'(\mu)\big|_\mathcal{E} \tag{C.11}$$

and $\tilde{g}_\mu : \mathcal{E} \mapsto R$ be defined as

$$\tilde{g}_\mu = \left\langle \mathrm{d}L\left(\int_\Theta \Phi \mathrm{d}\mu\right), \Phi(\xi)\right\rangle \tag{C.12}$$

so that for all $\xi \in \mathcal{E}$, $g_\mu(\xi) = \tilde{g}_\mu(\xi) + V(\xi)$. We establish a bound on the variations of $\tilde{g}$.

Lemma C.6 ([31, Lemma C.2.]). *For all $C_0 > 0$, there exists $M > 0$ such that for all $\mu, \nu \in \mathcal{M}(\Theta)$ such that $\|h(\mu)\|_{\mathrm{BL}}, \|h(\nu)\|_{\mathrm{BL}} < C_0$, it holds*

$$\|\tilde{g}_\mu - \tilde{g}_\nu\|_{C^1} \leq M \left\|\Phi\big|_{\mathcal{E}}\right\|_{C^1}^2 \cdot \|h(\mu) - h(\nu)\|_{\mathrm{BL}}, \tag{C.13}$$

where $\|\cdot\|_{\mathrm{C}^1}$ denotes the C^1 norm of a function defined as

$$\|\psi\|_{C^1} := \|\psi\|_\infty + \|\nabla\psi\|_\infty \quad \textit{for } \psi\colon \mathcal{E} \to \mathbb{R}.$$

Also, $\|\cdot\|_{\mathrm{BL}}$ denotes the bounded Lipschitz norm of a measure on $\mathcal{E}$ defined as

$$\|\nu\|_{\mathrm{BL}} := \sup\left\{\int_{\mathcal{E}} \psi \mathrm{d}\nu \,\middle|\, \psi\colon \mathcal{E} \to \mathbb{R}, \, \mathrm{Lip}(\psi) \leq 1, \, \|\psi\|_\infty \leq 1\right\} \quad \textit{for } \nu \in \mathcal{M}_+(\mathcal{E}),$$

where $\mathrm{Lip}(\psi)$ *is the smallest Lipschitz constant of ψ.*

C.2.2 Proof of Proposition 5.11

Let $\eta^* := \min_{\xi\in\mathcal{E}} g_\mu(\xi) < 0$ be the minimum that is attained on the compact set $\mathcal{E}$ and must be lower bounded $\eta^* > -\infty$. By Morse-Sard lemma [1] and the fact that Φ is smooth, there exists a regular value $-\eta \in]\eta^*, \eta^*/2]$ of g_μ, where $\eta > 0$. Let $K := g_\mu^{-1}(]-\infty, -\eta]) \subset \mathcal{E}$ be the $(-\eta)$-sublevel set of the regular value, and construct the subset $P := \pi_{\mathcal{E}}^{-1}(K) \subset \Theta$. As $\|h(\mu) - h(\mu_{t_0})\|_{BL} < \epsilon$, $\mu_{t_0}(P) > 0$ for $\epsilon > 0$ small enough.

By the regular value theorem, the boundary ∂K of K is a differentiable orientable compact submanifold of $\mathcal{E}$ of codimension 1. By definition of K and the regular value theorem, there exists $\beta > 0$ such that

$$\forall \xi \in \partial K: \quad \nabla g_\mu(\xi) \cdot n_\xi > \beta > 0, \tag{C.14}$$

where n_ξ is the unit normal vector to ∂K pointing outwards. Let $t_1 \in [t_0, \infty]$ the first time such that $\|h(\mu_{t_1}) - h(\mu)\|_{BL} \geq \epsilon$. The triangle inequality implies that $h(\mu_t)(\mathcal{E})$ is uniformly bounded from above for $t < t_1$.

For the sake of contradiction, suppose that $t_1 = \infty$. Without loss of generality, we let $t_0 = 0$. Consider now the flow $X : \mathbb{R}_+ \times \Theta \to \Theta$ defined for all $u \in \Theta$

$$X_0(u) = u \quad \text{and} \quad \partial_t X_t(u) = -\nabla F'(\mu_t)(X_t(u)), \tag{C.15}$$

which by [5, Lemma 8.1.4] is well-posed and unique. Notice also that μ_t can be thought of as the pushforward measure of μ_0 by X_t, that is $\mu_t = (X_t)_\# \mu_0$.

For $u_0 \in P$, define $(u_t)_{t\geq 0} := (X_t(u_0))_{t\geq 0} \subset \Theta$. We first establish that if

$$0 < \epsilon < \min\left(\beta, \frac{\eta}{2}\right) \cdot \left(2M \left\| \Phi|_{\mathcal{E}} \right\|_{C^1}^2\right)^{-1}, \tag{C.16}$$

then $(u_t)_{t\in[0,t_1]} \subset P$. To do this it suffices to show that if $u_t \in \partial P = \pi_{\mathcal{E}}^{-1}(\partial K)$, the gradient flow pushes u_t back inside P. Formulated on $K \subset \mathcal{E}$, it suffices to show that if $\xi = \pi_{\mathcal{E}}(u_t) \in \partial K$, then the angle between the gradient field $-\nabla g_{\mu_t}(\xi)$ and the unit normal vector to ∂K pointing outwards n_ξ is bigger than $\frac{\pi}{2}$. This is an easy consequence of Lemma C.6:

$$\begin{aligned} -\nabla g_{\mu_t}(\xi) \cdot n_\xi &= -\nabla g_{\mu}(\xi) \cdot n_\xi + (\nabla g_\mu(\xi) - \nabla g_{\mu_t}(\xi)) \cdot n_\xi \\ &\leq -\nabla g_{\mu}(\xi) \cdot n_\xi + \|g_\mu - g_{\mu_t}\|_{C_1} \\ &= -\nabla g_{\mu}(\xi) \cdot n_\xi + \|\tilde{g}_\mu - \tilde{g}_{\mu_t}\|_{C_1} \\ &\leq -\beta + \beta/2 = -\beta/2, \end{aligned}$$

where the first inequality holds by Cauchy-Schwarz and by the definition of $\|\cdot\|_{C^1}$, the second inequality holds by (C.14) and (C.16) with Lemma C.6. Hence $(u_t)_{t\in[0,t_1]} \subset P$. From (C.16) and by definition of K, we also deduce that for all $\xi \in K$,

$$g_{\mu_t}(\xi) = g_\mu(\xi) + (g_{\mu_t}(\xi) - g_\mu(\xi)) \leq -\eta/2. \tag{C.17}$$

We now look at the evolution of $\|u_t\|_A$ in time t, where A is the diagonal matrix with entries $A_{i,i} = \alpha_i$. As $F'(\mu_t)$ is (α, k) −homogeneous we have that

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} \|u_t\|_A^2 &= 2\left\langle \partial_t u_t,\, Au_t \right\rangle \\ &= 2\left\langle -\nabla F'(\mu_t)(u_t),\, Au_t \right\rangle \\ &= -2kF'(\mu_t)(u_t) \\ &= -2k\lambda(u_t)^k g_{\mu_t}(\pi_{\mathcal{E}}(u_t)) \\ &\geq k\eta\lambda(u_t)^k, \end{aligned} \tag{C.18}$$

where the third equality follows from Remark C.4 and the inequality follows from (C.17) as $\pi_{\mathcal{E}}(u_t) \in K$. We readily see that $t \mapsto \|u_t\|_A$ is increasing. Now, choose $c_0 > 0$ such that for $U_0 \coloneqq \{u \in P : \|u\|_A > c_0\}$,

$$\mu_0\left(U_0\right) > 1/2 \cdot \mu_0(P).$$

As $0 \notin P$, $c_0 > 0$ always exists and is finite. Note that for $u \in \Theta$, if $\lambda(u) \leq 1$, then, as $\alpha_i \geq 1$ and $\pi_{\mathcal{E}}(u) \in \mathcal{E}$,

$$\|u\|_A^2 = \sum_{i=1}^{d} \alpha_i u_i^2 = \sum_{i=1}^{d} \alpha_i \lambda(u)^{2\alpha_i} \pi_{\mathcal{E}}(u)_i^2 \leq \lambda(u)^2. \tag{C.19}$$

Case 1: If $\lambda(u_0) \leq 1$ and $u_0 \in U_0$, we deduce that $u_t \in U_0$ for all $t \geq 0$, and by (C.18) and (C.19), we have for all $t > 0$ such that $\lambda(u_t) \leq 1$,

$$\frac{\mathrm{d}}{\mathrm{d}t} \|u_t\|_A^2 \geq k\eta \|u_t\|_A^k \geq k\eta c_0^k.$$

Hence, by integrating the above inequality with respect to t, we deduce that

$$\inf \{t > 0\colon \ \|u_t\|_A \geq 1\} =: T_0 < \left(k\eta c_0^k\right)^{-1}.$$

Case 2: On the other hand, if $\lambda(u_0) > 1$ and $u_0 \in U_0$, then $\|u_0\|_A > 1$ as well, so $\|u_t\|_A > 1$ as $t \mapsto \|u_t\|_A$ is increasing, so $\lambda(u_t) > 1$ as well. Therefore, by (C.18),

$$\frac{\mathrm{d}}{\mathrm{d}t} \|u_t\|_A^2 \geq k\eta \cdot \lambda(u_t)^k \geq k\eta.$$

Integrating with respect to t yields $\|u_t\|_A^2 \geq \|u_0\|_A^2 + k\eta t$. Now, we deduce that similarly to (C.19), we have

$$\|u_t\|_A^2 = \sum_{i=1}^d \alpha_i (u_t)_i^2 = \sum_{i=1}^d \alpha_i \lambda(u_t)^{2\alpha_i} \pi_{\mathcal{E}}(u_t)_i^2 \leq \lambda(u_t)^{2\cdot \max_i \alpha_i}.$$

Hence, for $p := \frac{k}{\max_i \alpha_i} > 1$, we get

$$\lambda(u_t)^k \geq \|u_t\|_A^p \geq \left(\|u_0\|_A^2 + k\eta t\right)^{p/2}.$$

Finally, we estimate the mass of K under the measure $h(\mu_t)$. First note that by definition of h and by definition of u_t,

$$\begin{aligned} h(\mu_t)(K) &= \int_\Theta \lambda(u)^k \ \mathbb{1}\left(\pi_{\mathcal{E}}(u) \in K\right) \mathrm{d}\mu_t(u) \\ &= \int_\Theta \lambda(u_t)^k \ \mathbb{1}\left(\pi_{\mathcal{E}}(u_t) \in K\right) \mathrm{d}\mu_0(u_0) \\ &\geq \int_P \lambda(u_t)^k \mathrm{d}\mu_0(u_0) \end{aligned}$$

where the last inequality follows from P being stable under X_t. Now, for $t > T_0$,

$$\begin{aligned} h(\mu_t)(K) &\geq \int_{\substack{u_0 \in U_0 \\ \lambda(u_0) > 1}} \lambda(u_t)^k \mathrm{d}\mu_0(u_0) + \int_{\substack{u_0 \in U_0 \\ \lambda(u_0) \leq 1}} \lambda(u_t)^k \mathrm{d}\mu_0(u_0) \\ &\geq \int_{\substack{u_0 \in U_0 \\ \lambda(u_0) > 1}} \left(\|u_0\|_A^2 + k\eta t\right)^{p/2} \mathrm{d}\mu_0(u_0) + \int_{\substack{u_0 \in U_0 \\ \lambda(u_0) \leq 1}} \left(\|u_{T_0}\|_A^2 + k\eta(t - T_0)\right)^{p/2} \mathrm{d}\mu_0(u_0) \\ &\geq \frac{1}{2} \left(1 + k\eta(t - T_0)\right)^{p/2} \mu_0(P), \end{aligned}$$

where the first inequality holds as $U_0 \subset P$, the second one holds by Case 1 and Case 2, and the third one holds as $\mu_0(U_0) > 1/2 \cdot \mu_0(P)$, $\|u_{T_0}\|_A = 1$ in Case 1, and $\|u_0\|_A > 1$ in Case 2.

As a consequence of the above lower bound, the mass of K under the measure $h(\mu_t)$ blows up to infinity as $t \to \infty$, as $\mu_0(P) > 0$. This is a contradiction to $\|h(\mu) - h(\mu_t)\|_{BL} < \epsilon$. Thus, t_1 is finite, completing the proof.

□

C.2.3 Bound on the evolution of gradient flow functionals

Lemma C.7. *Let $\psi \in C^1(\Theta, \mathbb{R})$ such that there exists $C > 0$ satisfying $\|\nabla\psi(u)\|^2 \leq C\psi(u)$ for all $u \in \Theta$ and $\int_\Theta \psi \mathrm{d}\mu_0 < \infty$. Then*

$$\int_\Theta \psi(u)\mathrm{d}\mu_t(u) \leq 2\int_\Theta \psi(u)\mathrm{d}\mu_0(u) + \frac{C}{2}F(\mu_0)$$

Proof. Define $g(t) := \int_\Theta \psi(u)\mathrm{d}\mu_t(u)$. From the (distributional) continuity equation (5.8) (see [5, Equation 8.1.4]), we have

$$\frac{\mathrm{d}}{\mathrm{d}t}g(t) = \int_\Theta \nabla\psi(u)^\top v_t(u)\mathrm{d}\mu_t(u)$$

Further, Cauchy-Schwary inequality and the hypothesis yields

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}g(t) &\leq \left(\int_\Theta \|\nabla\psi(u)\|^2 \mathrm{d}\mu_t(u)\right)^{1/2}\left(\int_\Theta \|v_t(u)\|^2 \mathrm{d}\mu_t(u)\right)^{1/2} \\ &\leq C^{1/2}g(t)^{1/2}\left(-\frac{\mathrm{d}}{\mathrm{d}t}F(\mu_t)\right)^{1/2}.\end{aligned}$$

The last inequality is due to the conservation of energy for the Wasserstein gradient flow [5, Theorem 11.2.1.]. Therefore,

$$\frac{\mathrm{d}}{\mathrm{d}t}g(t)^{1/2} = \frac{1}{2}g(t)^{-1/2}\frac{\mathrm{d}}{\mathrm{d}t}g(t) \leq \frac{1}{2}C^{1/2}\left(-\frac{\mathrm{d}}{\mathrm{d}t}F(\mu_t)\right)^{1/2}$$

By integrating and Jensen's inequality, we deduce

$$g(t) = \left(g(0)^{1/2} + \int_0^t \frac{\mathrm{d}}{\mathrm{d}s}g(s)^{1/2}\right)^2 \leq 2g(0) + \frac{C}{2}\left(F(\mu_0) - F(\mu_t)\right) \leq 2g(0) + \frac{C}{2}F(\mu_0).$$

□

C.3 Proofs of Section 5.4

We prove in the section some of the intermediary results needed to derive the generalization bound for the Wasserstein gradient flow.

C.3.1 Proof of Lemma 5.20

An equivalent formulation of (5.29) is

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{S^{d-1}}\psi(\xi)\mathrm{d}\rho_t=\int_{S^{d-1}}\big[\mathrm{D}_\xi\psi\circ v_t^S(\xi)-2\psi(\xi)F'(M_0\rho_t)(\xi)\big]\mathrm{d}\rho_t(\xi),\quad \forall\psi\in C^\infty(S^{d-1}),$$

in the sense of distributions in $(0,T)$. It corresponds to the choice $\varphi(t,x)=\eta(t)\psi(x)$ in (5.9), with $\eta\in C_c^\infty(0,T)$. The equivalence comes from the integration part formula, and the fact that the linear span of seperable functions φ is dense in $C_c^\infty((0,T)\times S^{d-1})$. Fix any $\psi\in C^\infty(S^{d-1})$. We first compute the time-derivative of $\rho_t=M_0^{-1}h(\mu_t)$ using the definition of h and the continuity equation (5.8).

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{S^{d-1}}\psi(\xi)\mathrm{d}\rho_t(\xi)&=M_0^{-1}\frac{\mathrm{d}}{\mathrm{d}t}\int_\Theta\|u\|^2\,\psi(\pi(u))\mathrm{d}\mu_t(u)\\&=M_0^{-1}\int_\Theta\big\langle\,\|u\|^2\,\nabla\psi(\pi(u))+\psi(\pi(u))\nabla(\|u\|^2),\,v_t(u)\big\rangle\mathrm{d}\mu_t(u),\end{aligned}\tag{C.20}$$

where $v_t=-\nabla F'(\mu_t)$ is the tangent vector field of μ_t and $\pi(u)=\|u\|^{-1}u$ is the projection onto the $d-$dimensional sphere S^{d-1}. Its derivative is given by $\nabla\pi(u)=\|u\|^{-1}I_d-\|u\|^{-3}uu^\top$, so that

$$\nabla\psi(\pi(u))=\mathrm{D}_{\pi(u)}\psi\circ\nabla\pi(u).$$

Therefore, we get

$$\begin{aligned}\langle\nabla\psi(\pi(u)),\,v_t(u)\rangle&=\mathrm{D}_{\pi(u)}\psi\circ\nabla\pi(u)\circ v_t(u)\\&=\mathrm{D}_{\pi(u)}\psi\circ\iota_{\pi(u)}^{-1}\left(\|u\|^{-1}v_t(u)-\|u\|^{-3}uu^\top v_t(u)\right)\\&=\mathrm{D}_{\pi(u)}\psi\circ\iota_{\pi(u)}^{-1}\big(v_t(\pi(u))+2F'(\mu_t)(\pi(u))\pi(u)\big)\end{aligned}\tag{C.21}$$

as v_t is 1–homogeneous. Recall that $\iota_\xi^{-1}\colon\mathrm{span}(n_\xi)^\perp\hookrightarrow T_\xi S^{d-1}$ is the embedding from the set of vectors in $\mathbb{R}^d$ orthogonal to the normal vector n_ξ to the sphere S^{d-1} at point ξ to the tangent space of S^{d-1} at ξ. Now, for any $\xi\in S^{d-1}$, we get

$$\begin{aligned}v_t^S(\xi)=-\mathrm{D}_\xi F'_{S^{d-1}}(M_0\rho_t)&=-\mathrm{D}_\xi\left(\left\langle\mathrm{d}L\left(\int_\Theta\Phi\mathrm{d}h(\mu_t)\right),\,\Phi\big|_{S^{d-1}}\right\rangle+V\big|_{S^{d-1}}\right)\\&=\left\langle\mathrm{d}L\left(\int_\Theta\Phi\mathrm{d}h(\mu_t)\right),\,-\mathrm{D}_\xi\Phi\big|_{S^{d-1}}\right\rangle-\mathrm{D}_\xi V\big|_{S^{d-1}}\\&=\iota_\xi^{-1}\Big(-\nabla F'(\mu_t)(\xi)+2F'(\mu_t)(\xi)\xi\Big)\end{aligned}$$

The last equality holds by Remark C.5, and $F'_{S^{d-1}}(h(\mu)) = F'(\mu)(\xi)\big|_{S^{d-1}}$. Therefore, using (C.21), we get

$$\langle \nabla\psi(\pi(u)),\, v_t(u)\rangle = \mathrm{D}_{\pi(u)}\psi \circ v_t^S(\pi(u)).$$

Plugging the last equality into (C.20), we deduce

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{S^{d-1}} \psi(\xi)\mathrm{d}\rho_t(\xi) &= M_0^{-1}\int_\Theta \left[\, \|u\|^2\, \mathrm{D}_{\pi(u)}\psi \circ v_t^S(\pi(u)) + 2\psi(\pi(u))u^\top v_t(u)\right]\mathrm{d}\mu_t(u)\\ &= M_0^{-1}\int_\Theta \|u\|^2 \left[\mathrm{D}_{\pi(u)}\psi \circ v_t^S(\pi(u)) - 2\psi(\pi(u))F'(\mu_t)(\pi(u))\right]\mathrm{d}\mu_t(u)\\ &= M_0^{-1}\int_{S^{d-1}} \left[\mathrm{D}_\xi\psi \circ v_t^S(\xi) - 2\psi(\xi)F'(\mu_t)(\xi)\right]\mathrm{d}h(\mu_t)(\xi)\\ &= \int_{S^{d-1}} \left[\mathrm{D}_\xi\psi \circ v_t^S(\xi) - 2\psi(\xi)F'(M_0\rho_t)(\xi)\right]\mathrm{d}\rho_t(\xi), \end{aligned} \tag{C.22}$$

which proves (5.29) and the fact that v_t^S is the tangent vector field of ρ_t. It remains to prove that $\rho_t \in \mathcal{P}_2(S^{d-1})$. First, $\rho_0(S^{d-1}) = M_0^{-1}h(\mu_0)(S^{d-1}) = 1$, and for $\psi \equiv 1$ in (C.22), we get

$$\frac{\mathrm{d}}{\mathrm{d}t}\rho_t(S^{d-1}) = -2\int_{S^{d-1}} F'(M_0\rho_t)(\xi)\mathrm{d}\rho_t(\xi) = 0$$

by definition of the linear functional derivative (5.5). Hence, $\rho_t(S^{d-1}) = 1$ for all $t \geq 0$. The finite second moment of ρ_t comes from the fact that $\mu_t \in \mathcal{P}_2(\Theta)$.

As noticed in [101, 57], the solution of the distributional PDE (5.29) is the gradient flow of the functional $F_{S^{d-1}}$ with respect to the Hellinger-Kantorovich metric (also called the Wasserstein-Fisher-Rao metric), i.e. $\partial_t\rho_t = -\nabla_{KFR}F_{S^{d-1}}(M_0\rho_t)$. It can be decomposed into two orthogonal components, $-\mathrm{div}(v_t^S\rho_t) \in T_{\rho_t}W_2(S^{d-1})$, and $2F'(M_0\rho_t)\rho_t \in T_{\rho_t}\mathcal{M}^+_{FR}(S^{d-1})$, where $\mathcal{M}^+_{FR}(S^{d-1})$ is the space of positive measures on S^{d-1} induced by the Fisher-Rao metric. Therefore, v_t^S is a tangent vector field to ρ_t with respect to W_2.

□

C.3.2 Proof of Lemma 5.21

By [5, Proposition 8.4.6] adapted to complete Riemannian manifolds, we have that for L^1-a.e. $t \geq 0$, the unique optimal transport map between m_t^j and m_{t+h}^j is given, up to first order in $h \geq 0$, by the exponential map along the tangent vector field v_t^j. That means,

$$\lim_{h\to 0}\frac{W_2\left(m_{t+h}^j,\, \exp_{\#}^{j,h} m_t^j\right)}{|h|} = 0,$$

where $\exp^{j,h} \colon M \to M$ is defined by $\exp^{j,h}(p) = \exp_p(hv_t^j(p))$. As a consequence, we know that

$$\lim_{h\to 0} \frac{W_2^2\big(\exp_{\#}^{1,h} m_t^1,\, \exp_{\#}^{2,h} m_t^2\big) - W_2^2\left(m_t^1, m_t^2\right)}{h}. \tag{C.23}$$

exists and is equal to $\frac{\mathrm{d}}{\mathrm{d}t} W_2^2\left(m_t^1, m_t^2\right)$. Let $\tau_t \in \Gamma_o(m_t^1, m_t^2)$ be any optimal transport plan between m_t^1 and m_t^2. We can thus construct the transport plan

$$\zeta_{t,h} \coloneqq \left(\exp^{1,h} \circ \pi_1,\, \exp^{2,h} \circ \pi_2\right)_{\#} \tau_t \in \Gamma\left(\exp_{\#}^{1,h} m_t^1,\, \exp_{\#}^{2,h} m_t^2\right)$$

to find an upper bound to (C.23) as follows:

$$\begin{aligned} W_2^2\big(\exp_{\#}^{1,h} m_t^1,\, \exp_{\#}^{2,h} m_t^2\big) &\leq \int_{M^2} d_M(p_1,p_2)^2 \mathrm{d}\zeta_{t,h}(p_1,p_2) \\ &= \int_{M^2} d_M\left(\exp_{p_1}(hv_t^1(p_1)),\, \exp_{p_2}(hv_t^2(p_2))\right)^2 \mathrm{d}\tau_t(p_1,p_2). \end{aligned}$$

The inequality holds by definition of the Wasserstein distance. Now, for each $v \in T_pM$, we have

$$d_M(\exp_p(hv),\, p')^2 = d_M(p,\, p')^2 - 2h\big\langle \dot\gamma_{p,p'}(0),\, v\big\rangle + \mathcal{O}(h^2)$$

Therefore, as we can take $\gamma_{p',p}(s) = \gamma_{p,p'}(1-s)$, we deduce

$$\begin{aligned} &W_2^2\big(\exp_{\#}^{1,h} m_t^1,\, \exp_{\#}^{2,h} m_t^2\big) \\ &\leq W_2^2\big(m_t^1,\, m_t^2\big) + 2h\int_{M^2} \left(\big\langle \dot\gamma_{p_1,p_2}(1),\, v_t^2(p_2)\big\rangle - \big\langle \dot\gamma_{p_1,p_2}(0),\, v_t^1(p_1)\big\rangle\right) \mathrm{d}\tau_t(p_1,p_2) + \mathcal{O}(h^2). \end{aligned}$$

We finish the proof of (5.30) by rearranging the terms and using (C.23). To prove the second part, notice first that

$$\|\dot\gamma_{p_1,p_2}(1) - \dot\gamma_{p_1,p_2}(0)\| = \left\|\int_0^1 \ddot\gamma_{p_1,p_2}(s)\mathrm{d}s\right\| \leq d_M(p_1,p_2)^2.$$

Hence, using $\|\dot\gamma_{p_1,p_2}(0)\| = d_M(p_1,p_2)$ and the fact that τ_t is an optimal transport plan between m_t^1 and m_t^2, we deduce

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}t} W_2^2\left(m_t^1, m_t^2\right) \\ &\leq 2\int_{M^2} \left[d_M(p_1,p_2)^2 \left\|v_t^2\right\|_\infty + d_M(p_1,p_2)\left\|v_t^1(p_2) - v_t^2(p_1)\right\|\right] \mathrm{d}\tau_t(p_1,p_2) \\ &\leq 2\left\|v_t^2\right\|_\infty W_2^2\left(m_t^1, m_t^2\right) + 2\int_{M^2} d_M(p_1,p_2)\left(\mathrm{Lip}_t^1 d_M(p_1,p_2) + \left\|v_t^1 - v_t^2\right\|_\infty\right) \mathrm{d}\tau_t(p_1,p_2). \end{aligned}$$

We conclude by Jensen inequality.

□

Appendix D

Auxiliary results

Lemma D.1. *For any $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$, we have*

$$\|AB\|_F \leq \|A\|_2 \|B\|_F .$$

Proof. Let $B = [b_1, \ldots, b_p]$ the columns of B. Then $\|B\|_F^2 = \sum_{i=1}^p \|b_i\|_2^2$. We use the fact that the spectral norm is compatible with the Euclidian norm to deduce

$$\|AB\|_F^2 = \sum_{i=1}^p \|Ab_i\|_2^2 \leq \sum_{i=1}^p \|A\|_2^2 \|b_i\|_2^2 = \|A\|_2^2 \|B\|_F^2 .$$

□

Lemma D.2. *Let $x \in \mathbb{R}^d$ and $\{A_k : k = 1, \ldots, L\} \subset \mathbb{R}^{d \times d}$ such that* $\max_k \|A_k\|_2 < 1$. *Then*

$$\left\| \left[\prod_{k=1}^L (I_d + A_k) \right] x \right\|_2 \geq \|x\|_2 \prod_{k=1}^L (1 - \|A_k\|_2).$$

Proof. First observe that for $A, B \in \mathbb{R}^{d \times d}$ and $x \in \mathbb{R}^d$, we have $\|ABx\|_2 \geq \sigma_{\min}(A) \|Bx\|_2$, where $\sigma_{\min}(A)$ is the smallest singular value of A. This is easy to see, as $\sigma_{\min}(A)^2$ is the smallest eigenvalue of $A^\top A$, so

$$\|ABx\|_2^2 = (Bx)^\top A^\top A(Bx) \geq \sigma_{\min}(A)^2 \|Bx\|_2^2 .$$

Observe also that for all $A \in \mathbb{R}^{d \times d}$ with $\|A\|_2 < 1$, we have $\sigma_{\min}(I_d + A) \geq 1 - \|A\|_2 > 0$. Indeed, there exists $v \in \mathbb{R}^d$ such that $\|v\|_2 = 1$ and $v^\top (I_d + A) v = \sigma_{\min}(I_d + A)^2$. Hence,

$$\sigma_{\min}(I_d + A) = \left(1 + v^\top A v\right)^{1/2} \geq (1 - \|A\|_2)^{1/2} \geq 1 - \|A\|_2 .$$

Combining these two facts, we deduce that

$$\left\| \left[\prod_{k=1}^L (I_d + A_k) \right] x \right\|_2 \geq \|x\|_2 \prod_{k=1}^L \sigma_{\min}(I_d + A_k) \geq \|x\|_2 \prod_{k=1}^L (1 - \|A_k\|_2).$$

□

Lemma D.3. *Let $f \in C^2(\mathbb{R}^p)$ satisfying* $\sup_{\|x\|_2<R} \|\nabla^2 f(x)\|_2 \leq H_\infty$ *for some* $H_\infty, R > 0$. *Then, for all $x \in \mathbb{R}^p$ such that $\|x - x_0\|_2 < R$,*

$$\Big| f(x) - f(x_0) - \langle \nabla_x f(x), x - x_0 \rangle \Big| \leq \frac{H_\infty}{2} \|x - x_0\|_2^2 .$$

Proof. We apply the fundamental theorem of calculus for line integrals between x_0 and x:

$$f(x) - f(x_0) = \int_0^1 \big\langle \nabla_x f(x_0 + t(x - x_0)), x - x_0 \big\rangle \mathrm{d}t.$$

Hence, by Cauchy-Schwartz inequality and by hypothesis,

$$\begin{aligned} \Big| f(x) - f(x_0) - \big\langle \nabla_x f(x_0), x - x_0 \big\rangle \Big| &\leq \int_0^1 \|\nabla_x f(x_0 + t(x - x_0)) - \nabla_x f(x_0)\|_2 \|x - x_0\|_2 \,\mathrm{d}t \\ &\leq \int_0^1 H_\infty \|t(x - x_0)\|_2 \|x - x_0\|_2 \,\mathrm{d}t \\ &= \frac{H_\infty}{2} \|x - x_0\|_2^2 . \end{aligned}$$

□

Lemma D.4 (Discrete Grönwall inequalities). *Let* $(u_n)_{n\in\mathbb{N}}, (v_n)_{n\in\mathbb{N}}, (w_n)_{n\in\mathbb{N}} \subset \mathbb{R}_{>0}$. *Then*

(i) If $e_{n+1} \leq u_n e_n + v_n$ for each $n \geq 0$, then

$$e_n \leq \left(\prod_{n'=0}^{n-1} u_{n'} \right) e_0 + \sum_{n'=0}^{n-1} \left(\prod_{n''=n'+1}^{n-1} u_{n''} \right) v_{n'} .$$

(ii) If $g_0 > 0$ and $0 < g_{n+1} \leq u_n g_n + w_n g_n^{1/2}$, then

$$g_n^{1/2} \leq \left(\prod_{n'=0}^{n-1} u_n^{1/2} \right) g_0^{1/2} + \frac{1}{2} \sum_{n'=0}^{n-1} \left(\prod_{n''=n'+1}^{n-1} u_{n''}^{1/2} \right) \frac{w_{n'}}{u_{n'}^{1/2}} .$$

The first inequality is well-known, but we give proofs for both, for the sake of completeness.

Proof. To prove (i), we start by defining $\widetilde{e}_n = \left(\prod_{n'=0}^{n-1} u_{n'} \right)^{-1} e_n$. Then,

$$\widetilde{e}_{n+1} - \widetilde{e}_n = \left(\prod_{n'=0}^{n} u_{n'} \right)^{-1} (e_{n+1} - u_n e_n) \leq \left(\prod_{n'=0}^{n} u_{n'} \right)^{-1} v_n .$$

Hence, summing over n, we get

$$e_n = \left(\prod_{n'=0}^{n-1} u_{n'}\right) \widetilde{e}_n \le \left(\prod_{n'=0}^{n-1} u_{n'}\right) \left(e_0 + \sum_{n'=0}^{n-1} \left(\prod_{n''=0}^{n'} u_{n''}\right)^{-1} v_{n'}\right)$$
$$= \left(\prod_{n'=0}^{n-1} u_{n'}\right) e_0 + \sum_{n'=0}^{n-1} \left(\prod_{n''=n'+1}^{n-1} u_{n''}\right) v_{n'}.$$

To prove (ii), we simply complete the square: $u_n g_n + w_n g_n^{1/2} \le u_n \left(g_n^{1/2} + \frac{w_n}{2u_n}\right)^2$. Hence,

$$g_{n+1}^{1/2} \le u_n^{1/2} g_n^{1/2} + \frac{w_n}{2u_n^{1/2}}.$$

We can thus apply part (i) to $e_n = g_n^{1/2}$ to deduce the result.

□

Lemma D.5 (Continuous Grönwall inequality)**.** *Let $u, v\colon \mathbb{R}_+ \to \mathbb{R}$ two integrable functions satisfying $v(t) \ge 0$ and $v(t) + u(t) \int_t^\infty v(r)\mathrm{d}r \ge 0$ for each $t \ge 0$. Let $g : \mathbb{R}_+ \to \mathbb{R}_+$ be a differentiable function that satisfy $g(0) = 0$ and $\frac{\mathrm{d}}{\mathrm{d}t} g(t)^2 \le 2u(t)g(t)^2 + 2v(t)g(t)$ for each $t \ge 0$. Then*

$$g(t) \le \exp\left(\int_0^t u(s)\mathrm{d}s\right) \int_0^t v(s)\mathrm{d}s.$$

Proof. The condition is equivalent to

$$g(t)\left(\frac{\mathrm{d}}{\mathrm{d}t} g(t) - u(t)g(t) - v(t)\right) \le 0. \tag{D.1}$$

Define $U := \{t : g(t) > 0\}$. As g is continuous, U is open with respect to the standard topology on $\mathbb{R}_+$. Therefore, there exists $a_n < b_n$, $n \in \mathbb{N}$ such that the intervals $I_n = (a_n, b_n)$ are disjoint, and $U = \bigcup_{n=0}^\infty I_n$. We deduce that if $t \notin U$, then $g(t) = 0$ by positivity of g. If $t \in U$, then there exists $n \in \mathbb{N}$ such that $a_n < t < b_n$. Now, for each $s \in I_n$, $g(s) > 0$, so by (D.1), $\frac{\mathrm{d}}{\mathrm{d}t} g(s) \le u(s)g(s) + v(s)$. Hence, $G\colon I_n \to \mathbb{R}$ defined by $G(s) = \exp\left(-\int_{a_n}^s u(r)\mathrm{d}r\right) g(s)$ satisfy

$$\frac{\mathrm{d}}{\mathrm{d}s} G(s) = \exp\left(-\int_{a_n}^s u(r)\mathrm{d}r\right)\left(\frac{\mathrm{d}}{\mathrm{d}s} g(s) - u(s)g(s)\right)$$
$$\le \exp\left(-\int_{a_n}^s u(r)\mathrm{d}r\right) v(s) \le v(s)$$

Therefore, $G(s) \le G(a_n) + \int_{a_n}^s v(r)\mathrm{d}r$. By continuity of g and because $g(0) = 0$, we have $g(a_n) = 0$ and thus

$$g(s) \le \exp\left(\int_{a_n}^s u(r)\mathrm{d}r\right) \int_{a_n}^s v(r)\mathrm{d}r =: C_s(a_n) \tag{D.2}$$

Finally, we prove that for each $s \geq 0$, the function $C_s\colon [0,s] \to \mathbb{R}$ defined in (D.2) is decreasing. Indeed,

$$\frac{\mathrm{d}}{\mathrm{d}t} C_s(t) = -\exp\left(\int_t^s u(r)\mathrm{d}r\right)\left(v(t) + u(t)\int_t^s v(r)\mathrm{d}r\right).$$

Now, if $u(t) \geq 0$, $v(t) + u(t)\int_t^s v(r)\mathrm{d}r \geq 0$. Otherwise, $u(t) < 0$ and $v(t) + u(t)\int_t^s v(r)\mathrm{d}r \geq v(t) + u(t)\int_t^\infty v(r)\mathrm{d}r \geq 0$ by hypothesis. Hence, $\frac{\mathrm{d}}{\mathrm{d}t}C_s(t) \leq 0$ and C_s is decreasing, for each $s \geq 0$. From (D.2), we deduce

$$g(s) \leq C_s(a_n) \leq C_s(0) = \exp\left(\int_0^s u(r)\mathrm{d}r\right)\int_0^s v(r)\mathrm{d}r.$$

□

Lemma D.6. *Let $(Y_t)_{t\in[0,T]} \subset \mathbb{R}^{d\times d}$ be a continuous semimartingale that can be decomposed as $\mathrm{d}Y_t = A_t\mathrm{d}t + \mathrm{d}M_t$, where A is a square-integrable adapted process, M is a continuous square-integrable martingale with quadratic variation $\mathrm{d}\left[M, M^\top\right]_t = Q_t\,\mathrm{d}t$, and $\sup_{0\leq t\leq T}\|Q_t\|_F < Q_\infty < \infty$, where Q_∞ is a deterministic constant. Let $(X_t)_{t\in[0,T]} \subset \mathbb{R}^{d\times d}$ be the unique solution to the linear matrix-valued SDE $\mathrm{d}X_t = X_t\mathrm{d}Y_t$, with X_0 being a deterministic non-zero matrix. Then, for each $p > 1$, there exists a constant $C \equiv C(p, d, Q_\infty, X_0, T)$ such that*

$$\mathbb{E}\left[\sup_{t\in[0,T]}\|X_t\|_F^p\right] \leq C\,\mathbb{E}\left[\exp\left(2p\int_0^T |\mathrm{tr}\,(A_s)|\,\mathrm{d}s\right)\right]^{1/2}.$$

Proof. We apply the multidimensional Ito formula and linearity of the trace operator to first get

$$\begin{aligned}\mathrm{d}\,\|X_t\|_F^2 = \mathrm{d}\,\mathrm{tr}(X_t^\top X_t) &= \mathrm{tr}\left(\mathrm{d}X_t^\top X_t + X_t^\top \mathrm{d}X_t + \mathrm{d}\left[X^\top, X\right]_t\right)\\ &= \mathrm{tr}\left(X_t^\top X_t\left(\mathrm{d}Y_t + \mathrm{d}Y_t^\top\right) + \mathrm{d}\left[X^\top, X\right]_t\right)\end{aligned}$$

Now, by cyclic permutation invariance of the trace, we have

$$\begin{aligned}\mathrm{tr}\left(\mathrm{d}\left[X^\top, X\right]_t\right) = \mathrm{tr}\left(\mathrm{d}\left[X, X^\top\right]_t\right) &= \mathrm{tr}\left(X_t\,\mathrm{d}\left[Y, Y^\top\right]_t X_t^\top\right)\\ &= \mathrm{tr}\left(X_t^\top X_t\,\mathrm{d}\left[Y, Y^\top\right]_t\right)\\ &= \mathrm{tr}\left(X_t^\top X_t Q_t\,\mathrm{d}t\right).\end{aligned}$$

Therefore,

$$\mathrm{d}\,\|X_t\|_F^2 = \mathrm{tr}\left(X_t^\top X_t\left(A_t + A_t^\top + Q_t\right)\right)\mathrm{d}t + \|X_t\|_F^2\,\mathrm{d}N_t, \tag{D.3}$$

where

$$N_t := \operatorname{tr}\left(\int_0^t \frac{X_s^\top X_s}{\|X_s\|_F^2}\left(\mathrm{d}M_s + \mathrm{d}M_s^\top\right)\right) \tag{D.4}$$

is a martingale with quadratic variation given by

$$[N]_t = \sum_{i_1,j_1,i_2,j_2} \int_0^t \|X_s\|_F^{-4}\left(X_s^\top X_s\right)_{i_1j_1}\left(X_s^\top X_s\right)_{i_2j_2} \mathrm{d}\left[(M+M^\top)_{i_1j_1}, (M+M^\top)_{i_2j_2}\right]_s$$

By the Kunita-Watanabe inequality,

$$\begin{aligned}
[N]_t &\leq \sum_{i_1,j_1,i_2,j_2}\left(\int_0^t \|X_s\|_F^{-4}\left(X_s^\top X_s\right)_{i_1j_1}^2 \mathrm{d}\left[(M+M^\top)_{i_1j_1}\right]_s\right)^{1/2} \cdot \\
&\qquad \left(\int_0^t \|X_s\|_F^{-4}\left(X_s^\top X_s\right)_{i_2j_2}^2 \mathrm{d}\left[(M+M^\top)_{i_2j_2}\right]_s\right)^{1/2} \\
&= \left(\sum_{i,j}\left(\int_0^t \|X_s\|_F^{-4}\left(X_s^\top X_s\right)_{ij}^2 \mathrm{d}\left[(M+M^\top)_{ij}\right]_s\right)^{1/2}\right)^2 \\
&\leq d^2 \sum_{i,j}\int_0^t \|X_s\|_F^{-4}\left(X_s^\top X_s\right)_{ij}^2 \mathrm{d}\left[(M+M^\top)_{ij}\right]_s \\
&\leq 4d^2 Q_\infty \int_0^t \|X_s\|_F^{-4}\left\|X_s^\top X_s\right\|_F^2 \mathrm{d}s \leq 4d^2 Q_\infty t. \qquad \text{(D.5)}
\end{aligned}$$

The second inequality follows from Cauchy-Schwarz. Now, by conditioning on $\|X_t\|_F > 0$ if necessary, we have by the Ito's formula and (D.3)

$$\begin{aligned}
\mathrm{d}\log\|X_t\|_F^2 &= \|X_t\|_F^{-2}\mathrm{d}\|X_t\|_F^2 - \frac{1}{2}\|X_t\|_F^{-4}\mathrm{d}\left[\|X\|_F^2\right]_t \\
&= \|X_t\|_F^{-2}\operatorname{tr}\left(X_t^\top X_t\left(A_t + A_t^\top + Q_t\right)\right)\mathrm{d}t + \mathrm{d}N_t - \frac{1}{2}\mathrm{d}[N]_t.
\end{aligned}$$

Hence, by integrating and taking the exponential, we get

$$\begin{aligned}
\|X_t\|_F^2 &= \|X_0\|_F^2 \exp\left(\int_0^t \|X_s\|_F^{-2}\operatorname{tr}\left(X_s^\top X_s\left(A_s + A_s^\top + Q_s\right)\right)\mathrm{d}s\right)\exp\left(N_t - \frac{1}{2}[N]_t\right) \\
&\leq \|X_0\|_F^2 \exp\left(\int_0^t \left|\operatorname{tr}\left(2A_s + Q_s\right)\right|\mathrm{d}s\right)\mathcal{E}(N)_t \\
&\leq \|X_0\|_F^2 \exp\left(dQ_\infty T\right)\exp\left(2\int_0^t \left|\operatorname{tr}\left(A_s\right)\right|\mathrm{d}s\right)\mathcal{E}(N)_t,
\end{aligned}$$

where $\mathcal{E}(N)_t := \exp\left(N_t - \frac{1}{2}[N]_t\right)$ denotes the stochastic exponential of N. The first inequality follows from $\operatorname{tr}(AB) \leq |\operatorname{tr}(A)|\,|\operatorname{tr}(B)|$. Therefore, for $p > 1$, by Cauchy-Schwarz,

$$\mathbb{E}\left[\sup_{t\in[0,T]}\|X_t\|_F^p\right] \leq \|X_0\|_F^p \exp\left(\frac{p}{2}dQ_\infty T\right)\mathbb{E}\left[\exp\left(2p\int_0^T \left|\operatorname{tr}\left(A_s\right)\right|\mathrm{d}s\right)\right]^{1/2}\mathbb{E}\left[\sup_{t\in[0,T]}|\mathcal{E}(N)_t|^p\right]^{1/2}. \tag{D.6}$$

Now, as $\mathbb{E}\left[\exp\left(1/2\,[N]_T\right)\right] \leq \exp\left(2d^2 Q_\infty T\right) < \infty$, Novikov condition implies that $(\mathcal{E}(N)_t)_{t\in[0,T]}$ is a (continuous) martingale. Therefore, by Doob's inequality, we get

$$\mathbb{E}\left[\sup_{t\in[0,T]} |\mathcal{E}(N)_t|^p\right] \leq \left(\frac{p}{p-1}\right)^p \mathbb{E}\left[|\mathcal{E}(N)_T|^p\right]$$

Finally, we use the definition of the stochastic exponential and Cauchy-Schwarz to obtain

$$\begin{aligned}
\mathbb{E}\left[|\mathcal{E}(N)_T|^p\right] &= \mathbb{E}\left[\exp\left(pN_T - \frac{p}{2}\,[N]_T\right)\right] \\
&\leq \mathbb{E}\left[\exp\left(pN_T - p^2\,[N]_T\right)\exp\left(\frac{2p^2-p}{2}\,[N]_T\right)\right] \\
&\leq \mathbb{E}\left[\mathcal{E}(2pN)_T\right]^{1/2}\mathbb{E}\left[\exp\left((2p^2-p)\,[N]_T\right)\right]^{1/2} \\
&\leq \exp\left(2(2p^2-p)d^2Q_\infty T\right).
\end{aligned}$$

We plug this last inequality into (D.6) to conclude the proof, with

$$C(p,d,Q_\infty,X_0,T) \coloneqq \left(\frac{p\,\|X_0\|_F^2}{p-1}\right)^{p/2}\exp\left(2p^2d^2Q_\infty T\right).$$

□