High-dimensional Limits in Artificial Neural Networks

by

Aleksandr Shevchenko

August, 2024

A thesis submitted to the
Graduate School
of the
Institute of Science and Technology Austria
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Committee in charge:

Francesco Locatello, Chair
Marco Mondelli
Dan Adrian Alistarh
Christoph Lampert
Emmanuel Abbe

ISTA
Institute of
Science and
Technology
Austria

The thesis of Aleksandr Shevchenko, titled *High-dimensional Limits in Artificial Neural Networks*, is approved by:

Supervisor: Marco Mondelli, ISTA, Klosterneuburg, Austria

Signature: ______________________________

Co-supervisor: Dan Adrian Alistarh, ISTA, Klosterneuburg, Austria

Signature: ______________________________

Committee Member: Christoph Lampert, ISTA, Klosterneuburg, Austria

Signature: ______________________________

Committee Member: Emmanuel Abbe, EPFL, Lausanne, Switzerland

Signature: ______________________________

Defense Chair: Francesco Locatello, ISTA, Klosterneuburg, Austria

Signature: ______________________________

Signed page is on file

ISTA Thesis, ISSN: 2663-337X

I hereby declare that this thesis is my own work and that it does not contain other people's work without this being so stated; this thesis does not contain my previous work without this being stated, and the bibliography contains all the literature that I used in writing the dissertation.

I accept full responsibility for the content and factual accuracy of this work, including the data and their analysis and presentation, and the text and citation of other work.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee, and that this thesis has not been submitted for a higher degree to any other university or institution.

I certify that any republication of materials presented in this thesis has been approved by the relevant publishers and co-authors.

Signature: ______________________________

Aleksandr Shevchenko
August, 2024

Signed page is on file

Abstract

In the modern age of machine learning, artificial neural networks have become an integral part of many practical systems. One of the key ingredients of the success of the deep learning approach is recent computational advances which allowed the training of models with billions of parameters on large-scale data. Such over-parameterized and data-hungry regimes pose a challenge for the theoretical analysis of modern models since "classical" statistical wisdom is no longer applicable. In this view, it is paramount to extend or develop new machinery that will allow tackling the neural network analysis under new challenging asymptotic regimes, which is the focus of this thesis.

Large neural network systems are usually optimized via "local" search algorithms, such as stochastic gradient descent (SGD). However, given the high-dimensional nature of the parameter space, it is a priori not clear why such a crude "local" approach works so remarkably well in practice. We take a step towards demystifying this phenomenon by showing that the landscape of the SGD training dynamics exhibits a few beneficial properties for the optimization. First, we show that along the SGD trajectory an over-parameterized network is dropout stable. The emergence of dropout stability allows to conclude that the minima found by SGD are connected via a continuous path of small loss. This in turn means that the high-dimensional landscape of the neural network optimization problem is provably not so unfavourable to gradient-based training, due to mode connectivity. Next, we show that SGD for an over-parameterized network tends to find solutions that are functionally more "simple". This in turn means that the SGD minima are more robust, since a less complicated solution will less likely overfit the data. More formally, for a prototypical example of a *wide* two-layer ReLU network on a $1\mathrm{d}$ regression task we show that the SGD algorithm is implicitly selective in its choice of an interpolating solution. Namely, at convergence the neural network implements a piece-wise linear function with the number of linear regions depending *only* on the amount of training data. This is in contrast to a "smooth"-like behaviour which one would expect given such a severe over-parameterization of the model.

Diverging from the generic supervised setting of classification and regression problems, we analyze an auto-encoder model that is commonly used for representation learning and data compression. Despite the wide applicability of the auto-encoding paradigm, the theoretical understanding of their behaviour is limited even in the simplistic shallow case. The related work is restricted to *extreme* asymptotic regimes in which the auto-encoder is either severely over-parameterized or under-parameterized. In contrast, we provide a tight characterization for the 1-bit compression of Gaussian signals in the challenging proportional regime, i.e., the input dimension and the size of the compressed representation obey the same asymptotics. We also show that gradient-based methods are able to find a globally optimal solution and that the predictions made for Gaussian data extrapolate beyond - to the case of compression of natural images. Next, we relax the Gaussian assumption and study more structured input sources. We show that the shallow model is sometimes agnostic to the structure of the data

which results in a Gaussian-like behaviour. We prove that making the decoding component slightly less shallow is already enough to escape the “curse” of Gaussian performance.

Acknowledgements

I would like to start by thanking my advisors, Marco and Dan, for giving me the freedom to find and pursue my research interests, and for their unconditional support and guidance during my PhD journey. I also would like to thank Christoph and Emmanuel for being part of my thesis committee, and for their valuable feedback on my PhD research. I thank Francesco Locatello for being the chair of my defence committee.

I was extremely lucky to be part of two research groups at ISTA, and, thus, had a “doubled” opportunity to hang out with more amazing researchers. I would like to thank every member of Mondelli and Alistarh groups. I have learned a lot from you and I wish you smooth sailing during your research careers.

I would like to especially thank Kevin for being an amazing PhD pal and collaborator, and having similar “brainworms” to myself. Same goes for Simone: I will miss a lot our daily “gym bro” interactions. I would like to thank Dorsa for alleviating the “nerdiness” of my existence. I would like to thank Jen: our research nagging sessions helped me not get lost in analysing abstract spherical horses in a vacuum and not lose connection with practical research questions. I thank Anastasiya for a lot of pleasant conversations and showing many cool places in Vienna. I would like to thank my parents and all my friends for being there for me (or at least remotely :D) during “not-so-bright” times, especially when I was continuously hitting the wall to solve life and research problems.

Funding resources. This project was partially supported by the 2019 Lopez-Loreta Prize and by the FWF DK VGSCO, grant agreement number W1260-N35. The candidate would like to thank the IST Distributed Algorithms and Systems Lab for providing the computational resources. Furthermore, the candidate acknowledges the support from the Scientific Service Units of the Institute of Science and Technology Austria through resources provided by Scientific Computing.

About the Author

Alex obtained his bachelor's degree in computer science at the Higher School of Economics (HSE) in Moscow. For one year, he was part of a master's joint program between HSE and Skoltech focused on statistical learning theory. During his undergraduate studies, he was also part of the Bayesian Methods Research group and a research assistant at the Samsung-HSE laboratory affiliated with the group. He joined ISTA in September 2019 under the joint supervision of Marco Mondelli and Dan Adrian Alistarh. During his PhD, he was broadly interested in the theoretical foundations of machine learning leaning towards the analysis of training dynamics of over-parameterized models and non-convex optimization in high-dimensional settings. His work, published in the International Conference on Machine Learning and the Journal of Machine Learning Research, was focused on developing a theoretical understanding of various phenomena observed in artificial neural networks under high-dimensional asymptotic regimes.

List of Collaborators and Publications

This thesis is based on the following first-author/equal contribution publications of Aleksandr Shevchenko. We provide a summary of the contributions of each author, referred to by their initials.

- Alexander Shevchenko and Marco Mondelli. Landscape connectivity and dropout stability of SGD solutions for over-parameterized neural networks. *International Conference on Machine Learning*, 2020
  - Chapter 3 is based on this publication
  - MM initially proposed the problem to AS along with related work, supervised AS, and proposed/made improvements for the formal proofs and suggested additional experiments
  - All authors participated in the discussions which led to the development of the main results: dropout-stability and mode connectivity in mean-field regime
  - AS developed/adapted and wrote the majority of technical bulk of the paper
  - AS performed all related numerical simulations
  - All authors contributed to the writing and revising of the final manuscript
- Alexander Shevchenko, Vyacheslav Kungurtsev, and Marco Mondelli. Mean-field analysis of piecewise linear solutions for wide ReLU networks. *Journal of Machine Learning Research*, 2022
  - Chapter 4 is based on this publication
  - All authors participated in the project-related discussions
  - MM initially proposed the problem of studying a spline-like behaviour of over-parameterized shallow models on a 1d regression task along with the related work
  - MM supervised AS throughout the project and proposed improvements regarding the technical part
  - AS developed and wrote the formal bulk of the paper
  - AS performed all related numerical simulations
  - All authors contributed to the writing and revising of the final manuscript
  - VK suggested some of the useful related work
- Aleksandr Shevchenko, Kevin Kögler, Hamed Hassani, and Marco Mondelli. Fundamental limits of two-layer autoencoders, and achieving them with gradient methods. *International Conference on Machine Learning*, 2023

- Chapter 5 is based on this publication
- AS and KK contributed equally to the project
- All authors participated in the project related discussions
- MM, HH and AS proposed to look for an improvement upon random feature encoding
- AS discovered the linear behaviour of population risk for straight-through 1-bit shallow auto-encoding during numerical simulations and KK confirmed the alignment with vector-AMP prediction, which formally established the project direction
- AS, KK and HH contributed to the development of the lower bound in the case of isotropic data for rate $r \leq 1$
- AS and KK closed the isotropic case for rate $r > 1$
- AS and KK formulated and proved the convergence result for weight-tied gradient flow in the case of rate $r \leq 1$
- HH came up with a first sketch for GD-min convergence, AS and KK improved and finished the rigorous version which required overcoming a few major technical challenges of error control
- AS proposed to look at the block form for the case of non-isotropic data and got the first preliminary bound for a special case of covariance matrix
- AS and KK discovered the water-filling behaviour of the related block ranks
- AS proposed a KKT-based iterative scheme to get a tight numerical estimate for the lower bound
- AS and KK established formal guarantees for the KKT scheme
- AS performed all related numerical simulations
- MM supervised AS and KK throughout the project and proposed improvements regarding the writing of the technical and exposition parts
- All authors contributed to the writing and revising of the final manuscript

- Kevin Kögler, Alexander Shevchenko, Hamed Hassani, and Marco Mondelli. Compression of structured data with autoencoders: provable benefit of nonlinearities and depth. *International Conference on Machine Learning*, 2024
  - Chapter 6 is based on this publication
  - KK and AS contributed equally to the project
  - All authors participated in the project related discussions
  - KK and AS discovered that a shallow autoencoder ignores the “structure” of an $\mathrm{i.i.d.}$ signal
  - KK developed and wrote the convergence result for GD-min on sparse Gaussian data
  - KK and AS discovered phase transition behaviour in the minimizer, AS computed the related critical value
  - AS translated the RI-GAMP iterates into a suitable decoding network architecture, and empirically showed that two steps are close enough to the Bayes optimal performance of vector-AMP

- AS derived and implemented the related vector-AMP state evolution for sparse Gaussian source
- AS derived and implemented optimal denoisers for non-linear decoding
- AS provided a first version of the main body
- AS performed the majority of the related numerical simulations
- KK implemented a binning scheme for an empirical estimate of the optimal denoiser, when closed form expression was unavailable
- Throughout the project MM supervised KK and AS and coordinated regular progress meetings and suggested experimental setups for a better exposition of the paper main results
- All authors contributed to the writing and revising of the final manuscript

Table of Contents

List of Figures

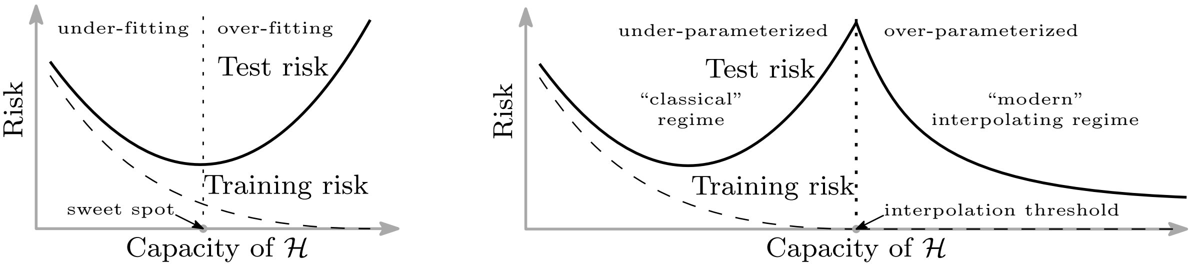
CHAPTER 1

Introduction

During the last decade, the field of machine learning experienced a drastic yet fruitful paradigm shift from classical statistical models, such as linear (kernel) models, to large-scale artificial neural networks. Such a rapid change, for the most part, was enabled by the advances on the computational side with the increased availability of graphical processing units (GPUs). The GPUs made possible training neural networks containing many millions or even billions of parameters on an extreme amount of data, which was impossible previously. Highly overparameterized functionally rich models trained on a vast amount of data are a key ingredient of success of neural networks in the modern age of deep learning.

The aforementioned paradigm allowed network models to achieve remarkable success on many diverse tasks, to name a few: image recognition [HZRS16, DBK^{+}21], machine translation [BCB14, VSP^{+}17], generative modeling and representation learning [KW14, GPAM^{+}14, HJA20, SCS^{+}22] and protein synthesis [JEP^{+}21]. However, despite the tremendous accomplishments of deep learning models, one might argue we have never been farther from understanding the "modus operandi" of systems that are used in practical machine learning tasks.

The aforementioned paradigm shift to large scale data-hungry systems resulted in a deviation from the standard asymptotic trade-offs between problem parameters. Namely, for a standard statistical analysis one usually assumes that the amount of available data is much larger comparatively to the complexity of the model (i.e., the number of model parameters). However, modern neural architectures do not operate under such scaling. The number of network parameters is usually close or even larger than the amount of data available. This difference in the modeling assumptions turns out to be crucial as a "classical statistical wisdom" no longer allows for a correct assessment of a neural network model properties. Perhaps the best example to corroborate the "failure" of the classical statistics approach is the phenomenon of *double descent* [BHMM19]. According to a well-established *bias-variance trade-off* the ability of a model to generalize beyond the training data is strongly dependent on the model "capacity". For simplicity of the exposition one might think of the number of network parameters as the main factor which determines the model capacity. Figure 1.1 (a) illustrates a U-shaped curve which encapsulates a classical understanding of the trade-off between generalization and the number of model parameters. Notably, if the model is too "over-parameterized" its generalization ability is hindered as too much flexibility will allow the model to overfit the data which will prevent it from capturing the "true" underlying structure. However, contrary to this common belief something more peculiar happens in practice for neural architectures.



(a) "Classical" statistics view

(b) Double descent characterization

Figure 1.1: Model generalization ability control via the capacity of a functional class: "classical" vs "modern" view. Illustration from [BHMM19].

The curve indeed obeys a U-shaped trend until a certain "interpolation threshold" point at which the model perfectly interpolates the data (reaching almost zero training error). After this "'breaking" point, the model generalization ability starts to improve eventually reaching a better value of the test risk. This is demonstrated in Figure 1.1 (b).

One of the possible explanations of the double descent phenomenon that has gained popularity recently is *benign overfitting* [BLLT20]. The high-level gist of benign overfitting may be summarized as follows: while the model certainly overfits the training data, the manner in which such overfitting occurs is not harmful to the model's generalization ability. Namely, the model overfits the "noise" component of the data which, due to certain data modelling assumptions, corresponds to the directions in the prediction space that are not important.

A somewhat orthogonal view suggests that when such over-parameterized models are "learned" via local search algorithms, for instance, stochastic gradient descent (SGD), the optimization procedure, despite the parameter redundancy, is biased towards more functionally "simple" and robust parameter configurations (see, e.g., [WTS$^+$19]). Crucially, such configurations are "easier" to find in the presence of over-parametrization. This is corroborated by the fact that smaller, "pruned" from the start, models fail to achieve the performance of their over-parameterized counterparts that are compressed at later stages of training [RFC20]. In contrast, the dynamic schemes (e.g., [PIVA21]) that allow for enough over-parameterization "slack" during the training perform remarkably well. These points motivate our study of over-parameterized models and, in particular, the analysis of the associated stochastic gradient descent solutions. Namely, we show that SGD solutions for over-parameterized neural networks exhibit a certain stability property. This property, in turn, implies that the SGD optimization landscape has a convenient structure, namely, the solutions found by SGD are connected via a continuous path along which the associated loss barely changes. Such "connectivity" of the stochastic gradient descent solutions might explain why for such high-dimensional models a rather simple local search algorithm yields remarkable results. Following up on the implicit bias of SGD dynamics, we prove that an over-parameterized two-layer ReLU network learns a much simpler data-interpolating solution than intuition would suggest. In particular, the network implements a piecewise linear function and the corresponding number of linear regions scales with the size of the dataset, but is independent of the number of network parameters.

The previously discussed challenges were mainly embedded in the context of supervised learning tasks, such as classification or regression problems. When it comes to practical applications of neural networks, unsupervised learning and, in particular, *representation learning* is hard to pass by. In this case, the most prominent neural architectural design is *auto-encoding*. Auto-encoders have achieved remarkable results in many sub-fields including, but not limited to, generative modeling [KW14] and data compression [TSCH17, AMT$^+$17]. However, the theoretical understanding of auto-encoder models is quite limited even for shallow architectures and is usually either limited to the case of linear activations [OSWS20] or extreme asymptotic regimes [Ngu21, RG22, CZ23]. To fill in the gaps, we study a prototypical problem of 1-bit compression via shallow auto-encoders, in the *proportional regime*. Under such regime, the input dimension and the dimension of the compressed representation obey the same asymptotics. For the case of Gaussian data we derive a tight lower bound on the compression performance of such neural architecture. Surprisingly, the predictions under the Gaussian data assumption go beyond and extrapolate extraordinarily well to the case of natural image data.

To explore further, we analyse a more structured *sparse* signal. Surprisingly, our theoretical analysis that is corroborated by numerics shows that, depending on the distribution of inputs, a shallow model is unaware of the signal structure, which results in Gaussian-like performance. However, for certain inputs the model indeed achieves better compression results, albeit the corresponding encoding is rather uninformative: the signal is passed through the encoding step with no changes (modulo an application of the 1-bit $\mathrm{sign}(\cdot)$ function). We characterize the property of the inputs that determines which of the related minimizers is selected. We also show that employing a deeper decoder without changing the shallow encoding structure already allows the model to provably improve upon Gaussian performance.

It is important to note that especially for the proportional (also referred to as "thermodynamic" in statistical physics) limit regime, the available theoretical understanding is quite scarce. In this view, the technical machinery developed for the analysis of the autoencoder model in Chapters 5 and 6 might be of a separate interest to the community.

Given the aforementioned scope of problems, the thesis is organized as follows:

- In Chapter 2, we give a brief outline of relevant machine learning concepts and techniques while simultaneously covering the existing terminology and notations.

- In Chapter 3, we study the SGD training of an over-parameterized fully-connected network under the mean-field regime [MMN18, AOY19]. We show that solutions found by SGD for such over-parameterized model are *dropout-stable* [KWL$^+$19] which implies that the related minima are connected. We validate our theoretical findings on the CIFAR-10 and MNIST classification tasks. The numerical simulations show remarkable agreement between the proposed theoretical analysis and practical evaluation. This chapter is based on our published work [SM20].

- In Chapter 4, we study the interpolating solutions of a regression problem for an extremely over-parameterized two-layer ReLU network found by noisy stochastic gradient descent. Taking a mean-field view once more, we analyze the properties of the limiting solution of the SGD dynamics directly via accessing its characterization that has a Gibbs form. Namely, we show that the curvature of the SGD solution vanishes almost everywhere except at a specific "cluster" set of points which can be uniquely identified. This observation in conjunction with the coupling described in [MMN18] allows us to dissect the functional form of the SGD solution. In particular, the network implements a piecewise linear function: the number of tangent changes is independent of the network size and scales linearly with the number of training samples. Remarkably, the described behaviour is significantly different from related works [WTS$^+$19, BGVV20]. Of separate interest, with our analysis we provide evidence that examining a Gibbs form directly, in some cases, might lead to a surprisingly tight characterization. This chapter is based on our published work [SKM22].

- In Chapter 5, we analyse 1-bit compression of Gaussian signal via a non-linear two-layer autoencoder model. We provide a tight lower bound (or asymptotically tight) on the population risk achieved by a shallow autoencoder for the case of *isotropic* Gaussian data. We also characterize the related minimizers (or minimizing sequence that saturates the bound in the thermodynamical limit). Given that empirically the "weight-tied" structure of encoding-decoding pair is optimal we derive the lower bound and the corresponding minimizing sequence for *non-isotropic* data (Gaussian with general covariance). In addition, we prove that two gradient-based optimization algorithms converge to the optimal solutions for the case of *isotropic* data. Validating the "Gaussian prediction" on natural data (MNIST, CIFAR-10) displays a remarkable agreement between the proposed lower bounds and the numerical simulation. This chapter is based on our published work [SKHM23].

- In Chapter 6, we go beyond the Gaussian design and consider the compression of a more structured signal. For a prototypical example of *sparse* Gaussian inputs we show that a shallow auto-encoder model is not capable of capturing the structure (sparsity) of the input signal. For a general i.i.d. signal, we observe a phase transition in the optimal solution that depends on a certain statistic of the input. Namely, the optimal solution switches from *random* orthogonal design, that results in no improvement upon Gaussian prediction, to a *deterministic* sparse solution that corresponds to a permutation of an identity matrix. We extensively validate our conjecture on a large family of data distributions. We show that enriching the expressive power of the decoder, by either adding a suitable non-linearity or adopting a "deeper" decoding architecture, provably allows to break the "curse" of Gaussian performance. In addition, we corroborate our findings with numerics on natural MNIST and CIFAR-10 images along with particle physics dataset from [YM21b]. This chapter is based on our published work [KSHM24].

- Lastly, in Chapter 7 we take a look back at our results and summarize possible directions for future research.

CHAPTER 2

Background

In this chapter, we introduce the basic notation and background for the training of neural network models. This brief summary will serve as a starting point for the discussions in the subsequent chapters. We will open with basic terminology for supervised learning and parametric models, namely, empirical/population risk minimization and how to find the corresponding minimizers via a local search algorithm, e.g., stochastic gradient descent (SGD). Later on, we discuss the *mean-field* framework [MMN18] for the analysis of training dynamics of neural networks (NNs), focusing on the case of a two-layer network architecture. We also cover a few related concepts for autoencoder models, including a variation of approximate message passing algorithms [VKM22], which find wide application in signal recovery (akin to the decoding step of autoencoding). We will keep the discussion in this chapter mostly informal and cover the necessary concepts in the subsequent chapters with more rigour if needed.

2.1 Parametric Models and Neural Networks

2.1.1 Parametric Models

Supervised learning. Supervised learning serves as the backbone of a substantial number of applied machine learning problems. In this view, it is necessary to establish the basic supervised learning principles right away. In the supervised learning context, one is given a set of training examples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^M$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ are inputs (e.g., a set of atmospheric measurements for an upcoming date) and $\boldsymbol{y}_i \in \mathbb{R}^K$ are the targets (e.g., temperature). The examples are usually independent and identically distributed (i.i.d.) samples from the underlying data distribution $\mathbb{P}$ supported on $\mathbb{R}^d \times \mathbb{R}^K$, i.e., $(\boldsymbol{x}_i, \boldsymbol{y}_i) \overset{\text{i.i.d.}}{\sim} \mathbb{P}$. The objective is then to recover the hidden dependence between inputs $\boldsymbol{x}$ and outputs $\boldsymbol{y}$ (e.g., predict temperature given a set of atmospheric measurements) from the training data $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^M$. To do so, one usually selects a function $f : \mathbb{R}^d \to \mathbb{R}^K$ from a certain functional class $\mathcal{H}$ that "fits" the available data in the best way according to some distance metric $\ell : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}$. The aforementioned pipeline is often referred as *empirical risk minimization* (ERM), and may be formalized as the following variational problem

$$f^* \in \arg\min_{f \in \mathcal{H}} \left\{ \frac{1}{M} \sum_{i=1}^{M} \ell(f(\boldsymbol{x}_i), \boldsymbol{y}_i) \right\}, \tag{2.1}$$

where f^* is the "best fit" candidate from $\mathcal{H}$. As the notation implies, the minimizer in (2.1) is not necessarily unique.

The most commonly used distances for supervised learning tasks are

- *squared loss* for regression tasks (whenever the target $\boldsymbol{y}$ is real-valued, akin to the temperature example):

$$\ell(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{K} \cdot \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2 = \frac{1}{K} \cdot \sum_{i=1}^{K} (\hat{y}_i - y_i)^2 , \tag{2.2}$$

  here the lower index stands for the i-th coordinate of $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ for the "predicted" value.

- *cross-entropy loss* for classification tasks (e.g., inputs $\boldsymbol{x}$ are flattened matrices/tensors which correspond to images of peaches and pears, and $\boldsymbol{y}$ is the correct one-hot encoded label, i.e., $\boldsymbol{y} = (1, 0)$ for peach and $\boldsymbol{y} = (0, 1)$ for pear):

$$\ell(\hat{\boldsymbol{y}}, \boldsymbol{y}) = - \sum_{j=1}^{k} y_i \cdot \log \hat{y}_i, \tag{2.3}$$

  where by convention $0 \cdot \log 0 = 0$ and $\hat{\boldsymbol{y}}$ stands for the predicted probabilities for each class, e.g., $\hat{\boldsymbol{y}} = (0.3, 0.7)$.

Except for a few experimental settings in Chapter 3, we mostly focus on regression-type problems and, hence, the main distance metric of interest for our purposes is squared loss.

While it is not the focus of the current thesis, it is important to outline a key concept of measuring the quality of the candidate f^*, which is usually done in terms of *generalization error*. Namely, the generalization error measures how well the ERM candidate f^* performs on unseen data (more precisely on all the data from $\mathbb{P}$), i.e.,

$$\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[\hat{\ell}(f^*(\boldsymbol{x}), \boldsymbol{y})], \tag{2.4}$$

for some distance function $\hat{\ell} : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}$, that does not necessarily coincide with the one of ERM in (2.1) for reasons we will highlight briefly later.

At this point, it is only appropriate to introduce the concept of population risk and the related population risk minimizer (PRM) which is widely used throughout the current thesis. In certain cases, by design, the medium has access to the whole data distribution $\mathbb{P}$. In this view, instead of choosing a predictor based on empirical risk

$$\frac{1}{M} \sum_{i=1}^{M} \ell(f(\boldsymbol{x}_i), \boldsymbol{y}_i), \tag{2.5}$$

one should aim to look for the minimizer of the *population risk*

$$\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[\ell(f(\boldsymbol{x}), \boldsymbol{y})]. \tag{2.6}$$

Consequently, the corresponding *population risk minimizer* f^* solves the following variational problem

$$f^* \in \arg\min_{f\in\mathcal{H}} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[\ell(f(\boldsymbol{x}), \boldsymbol{y})]. \tag{2.7}$$

Parametric models and gradient descent. The task of solving the variational problems in (2.1) and (2.7) in general spaces, e.g., L^2, is cumbersome and usually infeasible in the majority of practical applications. To account for that, one should exploit a family of parametric functions $\hat{\boldsymbol{y}}(\,\cdot\,, \boldsymbol{\Theta})$ which suits the application needs and allows for an efficient optimization (search of an optimal configuration $\boldsymbol{\Theta}$).

To be more concrete, fix some parametric form $\hat{\boldsymbol{y}}(\,\cdot\,, \boldsymbol{\Theta}) : \mathbb{R}^d \to \mathbb{R}^K$ and consider a related ERM problem, i.e.,

$$\frac{1}{M}\sum_{i=1}^{M} \ell(\hat{\boldsymbol{y}}(\boldsymbol{x}_i, \boldsymbol{\Theta}), \boldsymbol{y}_i) \to \min_{\boldsymbol{\Theta}}. \tag{2.8}$$

The most common way to solve (2.8) is to employ a version of a "local search" algorithm, such as gradient descent [BBV04] or higher-order Newton's method [Kel03] (assuming that (2.8) has the corresponding continuous derivatives). The methods of this type are referred as 'local search" algorithms since at each iteration k only a "local" information (order derivatives of (2.8) w.r.t. $\boldsymbol{\Theta}^k$) is used to get the next estimate $\boldsymbol{\Theta}^{k+1}$.

In this thesis, we focus on variations of gradient descent. The vanilla version of the gradient descent update with step size $\alpha > 0$ with a given initial choice of $\boldsymbol{\Theta}^0$ (e.g., random Gaussian initialization with constant order in dimension variance) is summarized as follows:

$$\boldsymbol{\Theta}^{k+1} = \boldsymbol{\Theta}^k - \alpha \cdot \nabla_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^k\right), \quad \nabla_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^k\right) = \nabla_{\boldsymbol{\Theta}}\left\{\frac{1}{M}\sum_{i=1}^{M} \ell(\hat{\boldsymbol{y}}(\boldsymbol{x}_i, \boldsymbol{\Theta}^k), \boldsymbol{y}_i)\right\}, \tag{2.9}$$

where $\nabla_{\boldsymbol{\Theta}}$ stands for the gradient of the corresponding function. Depending on the choice of gradient estimate $\nabla_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^k\right)$ one can distinguish a few versions of gradient descent:

- (*Stochastic* gradient descent). The gradient estimate is computed using a single random training data sample $(\widetilde{\boldsymbol{x}}_k, \widetilde{\boldsymbol{y}}_k)$, i.e.,

  $$\nabla_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^k\right) = \nabla_{\boldsymbol{\Theta}}\left\{\ell(\hat{\boldsymbol{y}}(\widetilde{\boldsymbol{x}}_k, \boldsymbol{\Theta}^k), \widetilde{\boldsymbol{y}}_k)\right\}, \quad (\widetilde{\boldsymbol{x}}_k, \widetilde{\boldsymbol{y}}_k) \overset{\text{i.i.d.}}{\sim} \frac{1}{M}\sum_{j=1}^{M} \delta_{(\boldsymbol{x}_j, \boldsymbol{y}_j)},$$

  here $\delta_{(\boldsymbol{x}, \boldsymbol{y})}$ stands for the delta distribution centered at $(\boldsymbol{x}, \boldsymbol{y})$.

- (*Batch* gradient descent). The gradient estimate is averaged across a batch of size B of the training data i.e.,

  $$\nabla_{\boldsymbol{\Theta}}\left(\boldsymbol{\Theta}^k\right) = \nabla_{\boldsymbol{\Theta}}\left\{\frac{1}{B}\sum_{i=1}^{B} \ell(\hat{\boldsymbol{y}}(\widetilde{\boldsymbol{x}}_{k,i}, \boldsymbol{\Theta}^k), \widetilde{\boldsymbol{y}}_{k,i})\right\},$$
  $$\{(\widetilde{\boldsymbol{x}}_{k,i}, \widetilde{\boldsymbol{y}}_{k,i})\}_{i=1}^{B} \overset{\text{i.i.d.}}{\sim} \frac{1}{M}\sum_{j=1}^{M} \delta_{(\boldsymbol{x}_j, \boldsymbol{y}_j)}.$$

The vanilla version in (2.9) is often referred as *full-batch* gradient descent, as batch size B is equal to the number of training samples M. There are variations to these schemes which mainly depend on how the corresponding examples are sampled. In particular, it is a common approach to substitute the i.i.d. sampling from the training data by doing a whole pass though the training set, i.e., for SGD:

$$(\widetilde{\boldsymbol{x}}_1, \widetilde{\boldsymbol{y}}_1) = (\boldsymbol{x}_1, \boldsymbol{y}_1), \quad (\widetilde{\boldsymbol{x}}_2, \widetilde{\boldsymbol{y}}_2) = (\boldsymbol{x}_2, \boldsymbol{y}_2), \quad \cdots,$$

and a similar scheme but for the batches of training examples in case of batch GD. A full pass through the dataset is often referred as an "epoch". It is also a common practice to randomly shuffle the order of training samples after each epoch to prevent the model from overfitting on a particular arrangement.

The main difference between the aforementioned variations of gradient descent is the trade-off between complexity/memory of the gradient estimate evaluation and its variance. In particular, one can establish upper bound guarantees (which depend explicitly on the variance) in the case of a convex (in $\mathbf{\Theta}$) ERM objective (2.8) (modulo additional requirements like L-smoothness and ℓ-strong convexity, which are of no particular interest to the current work, and, thus, omitted). Namely, assume that the variance of the gradient estimator $\nabla_{\mathbf{\Theta}}\left(\mathbf{\Theta}^k\right)$ is bounded (uniformly in iterations $k \in \mathbb{N}$) as follows:

$$\mathbb{E}\left[\left\|\nabla_{\mathbf{\Theta}}\left(\mathbf{\Theta}^k\right) - \nabla_{\mathbf{\Theta}}\left\{\frac{1}{M}\sum_{i=1}^{M}\ell(\hat{\boldsymbol{y}}(\boldsymbol{x}_i, \mathbf{\Theta}^k), \boldsymbol{y}_i)\right\}\right\|_2^2\right] \leq \sigma^2$$

(and that (2.8) is L-smooth and ℓ-strongly convex), then after $k = O\left(\ln \frac{\|\mathbf{\Theta}^0 - \mathbf{\Theta}^*\|_2^2}{\varepsilon}\right)$ iterations the following holds (see, for instance, [GLQ+19]):

$$\left\|\mathbf{\Theta}^k - \mathbf{\Theta}^*\right\|_2^2 \leq \varepsilon + \frac{\sigma^2}{\ell L}, \tag{2.10}$$

where $\varepsilon > 0$ (required precision) and $\mathbf{\Theta}^*$ is the unique minimizer (due to strong convexity) of the ERM (2.8). In particular, (2.10) suggests that the larger variance of the gradient estimate σ^2 will hinder the convergence speed of the corresponding stochastic gradient method.

In deep learning applications, the mini-batch SGD (mild sizes of batch) is the most common variation, as it provides a sufficient trade-off and arguably improves generalization compared to the full-batch version due to the presence of noise.

As mentioned before, it is also important to note that the metric $\hat{\ell}(\cdot,\cdot)$ used for generalization error might differ from the "surrogate" $\ell(\cdot,\cdot)$ that is used to obtain an optimal configuration $\mathbf{\Theta}^*$. Namely, a typical choice for a classification problem is the classification error defined as

$$\hat{\ell}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \mathbb{1}\left[\hat{\boldsymbol{y}} \neq \boldsymbol{y}\right], \tag{2.11}$$

which checks if the predicted label $\hat{\boldsymbol{y}}$ matches the ground truth $\boldsymbol{y}$. In this view, the problem of employing $\ell^*(\cdot,\cdot)$ directly for a gradient-based method is quite apparent - (2.11) is a non-smooth function of $\hat{\boldsymbol{y}}$, and, hence, of the model parameters $\mathbf{\Theta}$. That is why, in order to enable a gradient-based minimization one should employ a differentiable "surogate", which in the case of a classification task is typically cross-entropy (2.3).

For the purposes of the current thesis, it is important to highlight two versions of gradient descent that operate on the population risk (2.6). Namely, for the parametric family $\hat{\boldsymbol{y}}(\,\cdot\,, \mathbf{\Theta})$ we aim to find the corresponding minimizer of the population risk:

$$\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[\ell(\hat{\boldsymbol{y}}(\boldsymbol{x}, \mathbf{\Theta}), \boldsymbol{y})] \to \min_{\mathbf{\Theta}}. \tag{2.12}$$

In these terms, the corresponding version of gradient descent takes form

$$\mathbf{\Theta}^{k+1} = \mathbf{\Theta}^k - \alpha \cdot \nabla_{\mathbf{\Theta}}\left(\mathbf{\Theta}^k\right), \quad \nabla_{\mathbf{\Theta}}\left(\mathbf{\Theta}^k\right) = \nabla_{\mathbf{\Theta}}\left\{\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}}[\ell(\hat{\boldsymbol{y}}(\boldsymbol{x}, \mathbf{\Theta}), \boldsymbol{y})]\right\}. \tag{2.13}$$

The stochastic counterpart of (2.13) then corresponds to the estimate $\nabla_{\mathbf{\Theta}}\left(\mathbf{\Theta}^k\right)$ that is obtained using a single sample $(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{y}}_k)$ (or a batch of samples) from the data distribution $\mathbb{P}$, i.e.,

$$\nabla_{\mathbf{\Theta}}\left(\mathbf{\Theta}^k\right) = \nabla_{\mathbf{\Theta}}\left\{\ell(\hat{\boldsymbol{y}}(\tilde{\boldsymbol{x}}_k, \mathbf{\Theta}^k), \tilde{\boldsymbol{y}}_k)\right\}, \quad (\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{y}}_k) \sim \mathbb{P}. \tag{2.14}$$

In this view, it is quite natural to expect that under some regularity assumptions the scheme (2.14) should behave similarly to the "full" gradient version in (2.13). The version described in (2.14) is often referred as *online SGD*.

2.1.2 Neural Networks

The first occurrence of neural network models dates all the way back to the previous century [MP43]. However, artificial neural networks gained their popularity only recently with significant advances on the computation side enabled by graphical processing units (GPUs), which allowed for the training of deep network models at scale [KSH17] to showcase their capabilities across different domains [BYAV13, KW14, RMW14]. In this section, we describe a few neural network architectures of interest along with the relevant nuances.

Fully connected neural networks. We start with the simplest yet pioneering variation - a feed-forward neural network. In early literature, it is often referred as a multilayer perceptron (MLP) [Ros58], and, more recently, as a fully connected neural network (FCNN). In a nutshell, the FCNNs consist of sequential alternating layers of linear and fixed pointwise non-linear transformations. More formally, let $\boldsymbol{W}_1 \in \mathbb{R}^{N_1 \times d}$ and $\boldsymbol{b}_1 \in \mathbb{R}^{N_1}$ be the weights of the first "layer", $\boldsymbol{W}_i \in \mathbb{R}^{N_i \times N_{i-1}}$ and $\boldsymbol{b}_i \in \mathbb{R}^{N_i}$ for each $i \in \{2, \dots, L-1\}$ be the weights of the intermediate "layers", and define by $\boldsymbol{W}_L \in \mathbb{R}^{K \times N_{L-1}}$ with $\boldsymbol{b}_L \in \mathbb{R}^K$ the weights of the last "layer". Fix an *activation* function $\sigma : \mathbb{R} \to \mathbb{R}$. In these terms, a fully connected neural network implements the following composition:

$$\hat{\boldsymbol{y}}(\boldsymbol{x}, \mathbf{\Theta}) = \boldsymbol{W}_L \cdot \sigma(\boldsymbol{W}_{L-1} \cdot \sigma(\dots \boldsymbol{W}_2 \cdot \sigma(\boldsymbol{W}_1 \boldsymbol{x} + \boldsymbol{b}_1) + \boldsymbol{b}_2 \dots) + \boldsymbol{b}_{L-1}) + \boldsymbol{b}_L, \tag{2.15}$$

where $\mathbf{\Theta}$ stands for the collection of the model parameters $\{(\boldsymbol{W}_i, \boldsymbol{b}_i)\}_{i=1}^L$.

A few remarks regarding (2.15) are in order. First, note that one should choose $\sigma(\cdot)$ to be a non-linear function. The contrary will result in (2.15) implementing a linear transformation of the input $\boldsymbol{x}$, which is not sufficient to learn more complicated dependencies occurring in real world data. The importance of non-linear activations (which are also specifically non-polynomial) is also corroborated by a series of *universal approximation* results (see, for example, [Cyb89, Fun89, Bar93, LL20]). The universal approximation property (UAP) implies that any sufficiently regular function on a compact domain can be approximated arbitrarily well by a two-layer MLP (the network model in (2.15) with $L = 2$) once the layer width (N_1 in this case) is large enough with respect to the required precision. UAP is one of the main reasons to consider neural networks as a candidate for a parametric family of choice since it is capable of implementing (in theory, disregarding the optimal parameter search) practically any function of interest. It is also important to mention that for some tasks an additional activation (different from σ) may be applied to the outputs of (2.15). In particular, for a classification task, the network (2.15) outputs should correspond to the probabilities that are assigned for each of the K classes given an input $\boldsymbol{x}$. This means that $\hat{\boldsymbol{y}}(\boldsymbol{x}, \mathbf{\Theta})$ should belong to a $(K-1)$-dimensional simplex, formally:

$$\hat{\boldsymbol{y}}(\boldsymbol{x}, \mathbf{\Theta}) \in \Delta^{K-1} := \left\{\boldsymbol{u} \in \mathbb{R}_+^K : \sum_{j=1}^K u_j = 1\right\}. \tag{2.16}$$

To enforce the simplex condition 2.16, an extra *softmax* activation is applied

$$\mathrm{softmax}_\tau(\hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta})) := \frac{e^{-\tau\cdot\hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta})}}{\sum_{j=1}^{K} e^{-\tau\cdot\hat{y}(\boldsymbol{x},\boldsymbol{\Theta})_j}} \in \Delta^{n-1}, \tag{2.17}$$

where $\tau > 0$ stands for the temperature hyper-parameter, the exponent in the enumerator is applied entry-wise and $\hat{y}(\boldsymbol{x},\boldsymbol{\Theta})_j$ stands for the j-th entry of vector $\hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta})$. The temperature τ regulates how close $\mathrm{softmax}_\tau(\cdot)$ is to the $\mathrm{argmax}(\cdot)$ function. Namely, higher temperature results in a more uniform distribution, while for small values of τ, $\mathrm{softmax}_\tau(\cdot)$ puts more weight on the largest component of $\hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta})$ and

$$\lim_{\tau\to 0} \mathrm{softmax}_\tau(\hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta})) = \mathrm{argmax}(\hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta})).$$

While we do not use (2.17) in the theoretical analysis, it is a useful remark for the classification experiments considered in Chapter 3.

Backpropagation and residual connections. As discussed previously, the most common way to obtain an optimal solution for a parametric model $f_{\boldsymbol{\theta}}$ is to employ a local search algorithm such as SGD. The cornerstone of applying any gradient-based method is the computation of a gradient itself. A particular approach to do so for a neural network model (or any computational graph, for that matter[1]) is *backpropagation*, [RHW86] which is also referred as *backpropagation of error*. In a nutshell, backpropagation in this case is just a fancy name for applying chain-rule of a derivative. Formally, define the each layer outputs in the iterative fashion as follows:

$$\begin{aligned} \boldsymbol{a}_1(\boldsymbol{x}) &= \sigma(\boldsymbol{W}_1\cdot\boldsymbol{x}+\boldsymbol{b}_1), \\ \boldsymbol{a}_\ell(\boldsymbol{x}) &= \sigma(\boldsymbol{W}_\ell\cdot\boldsymbol{a}_{\ell-1}(\boldsymbol{x})+\boldsymbol{b}_\ell), \quad \ell\in\{2,\dots,L-1\}, \\ \boldsymbol{a}_L(\boldsymbol{x}) &= \boldsymbol{W}_L\cdot\boldsymbol{a}_{L-1}(\boldsymbol{x})+\boldsymbol{b}_L = \hat{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{\Theta}), \end{aligned} \tag{2.18}$$

where we suppressed the dependence of $\boldsymbol{a}_i$ on the corresponding layer parameters for a notation convenience. In these terms, we can access the derivatives during the “backward” pass[2] as follows: define the error derivative by

$$\mathrm{derr}(\boldsymbol{x}) := \frac{\partial}{\partial\boldsymbol{a}_L}\left[\ell(\boldsymbol{a}_L(\boldsymbol{x}),\boldsymbol{y})\right] \in \mathbb{R}^K \text{ (column vector)},$$

then unrolling (2.18) with chain-rule gives

$$\begin{aligned} &\nabla_{\boldsymbol{W}_L} = \mathrm{derr}(\boldsymbol{x})\cdot\boldsymbol{a}_{L-1}^\top(\boldsymbol{x}), && \nabla_{\boldsymbol{b}_L} = \mathrm{derr}(\boldsymbol{x}), && \nabla_{\boldsymbol{a}_{L-1}} = \boldsymbol{W}_L^\top\cdot\mathrm{derr}(\boldsymbol{x}) \\ &\nabla_{\boldsymbol{W}_\ell} = \nabla_{\boldsymbol{a}_\ell}\cdot\boldsymbol{a}_{\ell-1}^\top(\boldsymbol{x}), && \nabla_{\boldsymbol{b}_\ell} = \nabla_{\boldsymbol{a}_\ell}, && \nabla_{\boldsymbol{a}_{\ell-1}} = \boldsymbol{W}_\ell^\top\cdot\nabla_{\boldsymbol{a}_\ell}, \\ &\nabla_{\boldsymbol{W}_1} = \nabla_{\boldsymbol{a}_1}\cdot\boldsymbol{x}^\top, && \nabla_{\boldsymbol{b}_1} = \nabla_{\boldsymbol{a}_1}. \end{aligned} \tag{2.19}$$

It is now quite evident why the scheme in (2.19) is referred as a backpropagation of *error*, since the error derivatives $\mathrm{derr}(\boldsymbol{x})$ are propagated from the outputs to the network inputs. The algorithm in (2.19) can be implemented for any particular type of layer (not exclusive to the fully connected layer) as long as the corresponding transformation is differentiable, for

[1] Any left-to-right oriented directed graph without loops, where the leftmost set of nodes corresponds to the inputs and the rightmost to the outputs.

[2] So called “forward” pass constitutes in computing the activation quantities in (2.18).

instance, convolutional layers in convolutional neural networks (CNNs), which we will discuss briefly later.

Increasing the depth L of (2.15) results in improved representation capabilities. However, this comes at a cost. One of the important drawbacks of utilizing a deep MLP network in practice is the *vanishing gradient* problem. Namely, for deeper models the norm of the gradients that the first layers receives tends to be quite small due to multiplicative nature of the updates in (2.19). This issue is specifically more prominent for networks with hyperbolic activations such as hyperbolic tangent, i.e.,

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}},$$

since after a few iterations of scheme (2.19) the argument values tend to fall in the "flat" regions of activation, where the derivative is vanishing. [HZRS16] proposes to solve this issue (among a few others) with a particular modification to the network architecture - a *residual connection*. The concept of a residual connection is quite general and applies to any layer transformation $\psi : \mathbb{R}^{d_i} \to \mathbb{R}^{d_i}$. Namely, the idea of a residual connection is to modify $\psi(\cdot)$ as follows

$$\widetilde{\psi}(\boldsymbol{a}) = \psi(\boldsymbol{a}) + \alpha \cdot \boldsymbol{a}, \quad \alpha \in \mathbb{R}, \tag{2.20}$$

where α is either trained or is set to 1. The modification 2.20 aims to alleviate the "vanishing" gradients issue, since the fraction of signal $\boldsymbol{a}$ is always propagated during the backward pass:

$$\nabla_{\boldsymbol{a}} = \nabla_{\boldsymbol{a}} \left[\psi(\boldsymbol{a})\right] + \alpha \cdot \mathbf{1}.$$

One can notice that the dimension of the inputs after the "residual" layer (2.20) remains unchanged. In residual architectures, a common way of dimensionality reduction is to use predefined (not trainable) operations such as max/average pooling. Max pooling returns a maximum element in a sliding window of the input. For instance, consider a vector input $\boldsymbol{x} \in \mathbb{R}^{2d_i}$ to which we apply a max pooling with a sliding window of size 2 and stride 2, where stride defines by how much the corresponding sliding window is shifted:

$$\text{max\_pool}(\boldsymbol{x}) \in \mathbb{R}^{d_i}, \quad \text{max\_pool}(\boldsymbol{x})_j = \max\{x_{2j-1}, x_{2j}\}, \quad j \in \{1, \dots, d_i\}.$$

If the input size is not odd (or in general not compatible with a window/stride size) it is common to just "pad" the inputs $\boldsymbol{x}$ with zeros to get the argument of a consistent size. The working principle for the average pooling operation remains the same modulo the fact that instead of returning the maximal element it yields an average over the elements in the sliding window. It is also important to note that instead of fixed operations like the ones discussed previously one can use an adjustable version, e.g., weighted average for average pooling or simply another trainable linear transformation $W \in \mathbb{R}^{d_{i+1} \times d_i}$ with $d_{i+1} < d_i$. Similar techniques can be used for the upscaling of the input dimension that is commonly used in generative or decoding architectures [GPAM+14, KW14]. We limit the discussion of the upscaling since it bares little relevance for the current thesis.

The residual connection (2.20) is commonly used alongside an activation whose derivatives are less dampened at one of the extremes. A particularly common choice for such activation is rectified linear unit (ReLU), also know as "hard-plus":

$$\text{ReLU}(x) = (x)_+ = \max\{0, x\}, \tag{2.21}$$

or its truncated version (usually at value 5, justified by empirics)

$$\text{ReLU5}(x) = \mathbb{1}\{x \leq 5\} \cdot \text{ReLU}(x) + \mathbb{1}\{x > 5\} \cdot 5,$$

or the "leaky" variation

$$\mathrm{LeakyReLU}(x) = \mathbb{1}\{x < 0\} \cdot \alpha + \mathbb{1}\{x \geq 0\} \cdot x, \quad \alpha < 0.$$

While activation in (2.21) is strictly speaking non-smooth, a particular choice of the elements from the sub-gradient, i.e.,

$$\nabla_x[\mathrm{ReLU}(x)] = \mathbb{1}\{x \geq 0\},$$

works well in practice to enable gradient-based optimization.

Convolutional networks. We now briefly highlight the convolutional network architecture used in a few image data experiments in Chapter 3. We start by defining a two-dimensional convolution. Let $\boldsymbol{x} \in \mathbb{R}^{\mathrm{width} \times \mathrm{height}}$ be a matrix-valued input (for instance, a single channel of an RGB image) and consider convolutional kernel $\boldsymbol{\theta} \in \mathbb{R}^{k_{\mathrm{width}} \times k_{\mathrm{height}}}$ with *strides* $(s_{\mathrm{width}}, s_{\mathrm{height}})$. Then an application of a convolution $T_{\boldsymbol{\theta}}$ to the input $\boldsymbol{x}$ is formalized as follows:

$$T_{\boldsymbol{\theta}}(\boldsymbol{x})_{i,j} := \sum_{k=1}^{k_{\mathrm{width}}} \sum_{\ell=1}^{k_{\mathrm{height}}} \theta_{k,\ell} \cdot x_{(i-1)\cdot s_{\mathrm{width}}+k,(j-1)\cdot s_{\mathrm{height}}+\ell}. \tag{2.22}$$

The operation in (2.22) can be intuitively viewed as a weighted average with weights $\boldsymbol{\theta}$ in a sliding window of size $k_{\mathrm{width}} \times k_{\mathrm{height}}$ which moves over the input $\boldsymbol{x}$ with step sizes $(s_{\mathrm{width}}, s_{\mathrm{height}})$ in the x and y-axis respectively. Usually, the convolutional layers are applied to a multi-channel input, i.e., $\boldsymbol{x} \in \mathbb{R}^{\mathrm{channels} \times \mathrm{width} \times \mathrm{height}}$. In this case the operation in (2.22) stays the same channel-wise. However, different channels have different weights, i.e., $\boldsymbol{\theta}$ corresponds to a tensor of size $\mathrm{channels} \times k_{\mathrm{width}} \times k_{\mathrm{height}}$, and the result of (2.22) is additionally averaged over the input channels. Typically, each of the convolutional layers has multiple filters $\boldsymbol{\theta} \in \mathbb{R}^{\mathrm{channels} \times k_{\mathrm{width}} \times k_{\mathrm{height}}}$. In this view, each convolutional layer output is a three-dimensional tensor with the first dimension which corresponds to the number of filters.

It is important to note that a convolution can be implemented as a linear layer with a matrix of a particular circular structure. However, due to the circular structure, the multiplication by such a matrix can be implemented much more efficiently than a general linear transformation. In addition, the number of parameters that a convolution requires is significantly smaller than that of a fully connected layer of a compatible size. This makes the search space of SGD much smaller than for a conventional fully connected network, which results in better overall performance. Nevertheless, it is important to note that such architectural "bias" usually comes with a strong connection to the structure of data. Namely, convolutional networks are arguably designed for data with strong local correlations (inline with a local nature of averaging in (2.22)), for example, image data. In this view, one should not expect such architectural choice to benefit the data with strong long-distance correlations, e.g., time series. Although, a more "dilated" version of a convolution may be used in this case. However, it is arguably substantially different from the (2.22) and bears no particular interest to the current thesis.

To summarize, a convolutional neural network has a compositional nature akin to (2.15) with sequential application of convolutional layers and non-linearities, and dimension reduction layers (the discussed earlier pooling operations). Typically, the output of the last convolutional layer is flattened to a vector format and then passed through the final linear layer responsible for classification/regression. The outputs of the second to last layer of a neural network are sometimes referred as "features".

Vision datasets, input preprocessing and augmentation. In the following paragraphs, we briefly outline natural image datasets used for the experiments in Chapters 3, 5, and 6, along with common input preprocessing and data augmentation techniques. We start with the dataset of handwritten digits - MNIST. It contains 60000 training and 10000 validation binary ($0-1$ valued) images of handwritten digits from 0 to 9 stored in matrix 28×28 format. Remarkably, the most famous early exposition of convolutional network capabilities (LeNet5 [LBBH98]) was demonstrated on the MNIST data. Alongside MNIST, the CIFAR-10 dataset is one of the commonly small-scale baselines used for neural network models. Similarly to MNIST, it is composed of 60000 training and 10000 validation images. However, the structure of images itself is less synthetic - CIFAR-10 data corresponds to the downscaled RGB (real-valued) images of size 32×32 of natural objects such as cats, dogs, trucks, etc. (10 classes in total). A smaller convolutional network, e.g., the aforementioned LeNet5, yields an unsatisfactory performance on more challenging CIFAR-10 data. It is then more common to use a variant of [HZRS16] or fully connected classifier stacked on top of pretrained on ImageNet [DDS$^+$09] "feature" extractor.

A common technique to make a gradient-based training of any parametric model more stable, and, consequently, more accurate is to use a variant of the data standardization technique. Here we are going to discuss a variation that is commonly used for vision tasks on MNIST and CIFAR-10 datasets. Given the training data inputs $\{\boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^M$ we compute their empirical mean and variance as follows

$$\mu = \frac{1}{Md} \cdot \sum_{i,j=1}^{M,d} x_{i,j}, \quad \sigma^2 = \frac{1}{Md-1} \sum_{i,j=1}^{M,d} (x_{i,j} - \mu)^2 \tag{2.23}$$

where $x_{i,j}$ stands for the j-th coordinate of i-th training sample, and the second fraction correction $Md-1$ is necessary to obtain an unbiased estimate (under $\mathrm{i.i.d.}$ hypothesis on the components of the input). Using the computed statistics (2.23) the inputs $\boldsymbol{x}_i$ are then transformed as such

$$T(\boldsymbol{x}_i)_j = \frac{x_{i,j} - \mu}{\sigma}. \tag{2.24}$$

For RGB images it is common to collect the related statistics (2.23) channel-wise, and then apply (2.24) to each channel independently. The same transformation (using training statistics) is then applied during the testing phase. The transformation (2.24) aims to enforce the data to have zero mean and unit variance which improves numerical stability (e.g., reduces the variance in the network activations, and, consequently, the gradient computation).

The parametric models and neural networks especially are known to benefit from more training data available (e.g., in terms of the generalization error). A popular approach is to artificially inflate the training data via augmentation techniques. We are going to briefly describe a few of them that are commonly used for vision tasks. The first one corresponds to a random rotation of the initial image around its center (with a possible zero padding to preserve the original image size). The second one is random cropping: a smaller (rectangular) area of the original image is randomly selected instead of the whole image, and then either padded with zeros or upscaled to meet the initial input size. Another common input transformation is horizontal flip. In this case, the original input is reflected along y-axis passing through the image center. Usually, these augmentations are combined sequentially:

$$\mathrm{random\_crop} \to \mathrm{random\_rotation} \text{ and/or } \mathrm{random\_horizontal\_flip}.$$

2.2 Mean-field Framework

In this section, we briefly discuss the framework to describe the gradient dynamics of a two-layer neural network model in (2.15), which we base our analysis on in Chapters 3 and 4. We start by rewriting the model in (2.15) for the case of two layers and a single output (i.e., $L = 2$ and $K = 1$) in a more convenient form:

$$\begin{aligned} \hat{y}_N(\boldsymbol{x}, \boldsymbol{\Theta}) &= \frac{1}{N} \cdot \boldsymbol{W}_2 \cdot \sigma(\boldsymbol{W}_1 \cdot \boldsymbol{x} + \boldsymbol{b}_1) \\ &= \frac{1}{N} \sum_{i=1}^{N} w_{2,i} \cdot \sigma(\boldsymbol{w}_{1,i}^\top \cdot \boldsymbol{x} + b_{1,i}) = \frac{1}{N} \sum_{i=1}^{N} \sigma^*(\boldsymbol{x}, \boldsymbol{\theta}_i), \end{aligned} \tag{2.25}$$

where $N = N_1$, $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i\}_{i=1}^N$ with $\boldsymbol{\theta}_i = (w_{2,i}, \boldsymbol{w}_{1,i}, b_{1,i})$ and σ^* stands for the modified activation

$$\sigma^*(\boldsymbol{x}, \boldsymbol{\theta}_i) := w_{2,i} \cdot \sigma(\boldsymbol{w}_{1,i}^\top \cdot \boldsymbol{x} + b_{1,i}). \tag{2.26}$$

In (2.25) we introduced additional scaling $1/N$ for reasons that will become apparent later and omitted the bias term $\boldsymbol{b}_2$ to enable rewriting in the RHS of (2.25). The extra lower index in $\hat{y}$ emphasizes the explicit dependence on the number of neurons N.

While the definition in (2.26) is necessary for the connection with the previously discussed multi-layer network (2.15), from now on we will consider a more generic choice of σ^* that is not restricted to (2.26). Formally, we consider $\sigma^* : \mathbb{R}^d \times \mathbb{R}^D \to \mathbb{R}$, where D stands for the ambient dimension of each neuron parameters $\boldsymbol{\theta}_i \in \mathbb{R}^D$. To emphasize, $\boldsymbol{\theta}_i$ is not restricted to the form $(w_{2,i}, \boldsymbol{w}_{1,i}, b_{1,i})$ described previously. Given this, we can now proceed with the description of the mean-field framework.

Notice that the RHS of (2.38) bears a strong resemblance to an average. Namely, define the *empirical* distribution of the network weights $\boldsymbol{\Theta}$ as follows

$$\hat{\rho}^N = \frac{1}{N} \sum_{j=1}^{N} \delta_{\boldsymbol{\theta}_j}. \tag{2.27}$$

In these terms, the equation in (2.38) can be further rewritten as an expectation over the empirical distribution of the network weights (2.27), i.e.,

$$\hat{y}_N(\boldsymbol{x}, \boldsymbol{\Theta}) = \mathbb{E}_{\boldsymbol{\theta} \sim \hat{\rho}^N} \left[\sigma^*(\boldsymbol{x}, \boldsymbol{\theta})\right] = \int \sigma^*(\boldsymbol{\theta}, \boldsymbol{x}) \hat{\rho}^N(\mathrm{d}\boldsymbol{\theta}).$$

Given that the distribution $\rho_0 \in \mathcal{P}(\mathbb{R}^D)$ from which the initial weights $\{\boldsymbol{\theta}_i^0\}_{i=1}^N$ are sampled, i.e.,

$$\left\{\boldsymbol{\theta}_i^0\right\}_{i=1}^N \overset{\text{i.i.d.}}{\sim} \rho_0,$$

is sufficiently regular, one would expect that at the initialization the network output concentrates to the mean taken with respect to ρ_0

$$\begin{aligned} &\hat{y}_N(\boldsymbol{x}, \boldsymbol{\Theta}^0) = \mathbb{E}_{\boldsymbol{\theta} \sim \hat{\rho}_0^N} \left[\sigma^*(\boldsymbol{\theta}, \boldsymbol{x})\right] \approx \int \sigma^*(\boldsymbol{\theta}, \boldsymbol{x}) \rho_0(\mathrm{d}\boldsymbol{\theta}) = y_{\rho_0}^{\sigma^*}(\boldsymbol{x}), \\ &\hat{\rho}_0^N = \frac{1}{N} \sum_{j=1}^{N} \delta_{\boldsymbol{\theta}_j^0}, \quad \boldsymbol{\Theta}^0 = \{\boldsymbol{\theta}_j^0\}_{j=1}^N, \end{aligned} \tag{2.28}$$

for a large enough number of weights N (that are samples for a mean estimation in this case). While this is quite remarkable, the question stands: if such a property is preserved throughout the network (2.25) gradient-based training and how it could be exploited.

Consider minimizing the regularized squared population risk

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}}\left[(\hat{y}_N(\boldsymbol{x},\boldsymbol{\Theta})-y)^2\right]+\frac{\lambda}{N}\sum_{i=1}^{N}\|\boldsymbol{\theta}_i\|_2^2 \tag{2.29}$$

via the *noisy* version of online SGD (2.14) with the gradient rescaled by a multiplicative factor of the number of neurons N:

$$\begin{aligned}\boldsymbol{\theta}_i^{k+1}&=(1-2\lambda s_k)\cdot\boldsymbol{\theta}_i^k+2s_k\cdot(\widetilde{y}_k-\hat{y}_N(\widetilde{\boldsymbol{x}}_k,\boldsymbol{\Theta}^k))\cdot\nabla_{\boldsymbol{\theta}_i}\left(\sigma^*(\widetilde{\boldsymbol{x}}_k,\boldsymbol{\theta}_i^k)\right)+\sqrt{\frac{2s_k}{\beta}}\cdot\boldsymbol{g}_i^k,\\ \boldsymbol{g}_i^k&\overset{\text{i.i.d.}}{\sim}\mathcal{N}(\boldsymbol{0},\boldsymbol{I}_D),\quad(\widetilde{\boldsymbol{x}}_k,\widetilde{y}_k)\overset{\text{i.i.d.}}{\sim}\mathbb{P},\quad i\in[N]:=\{1,\dots,N\},\quad k\in\mathbb{N},\end{aligned} \tag{2.30}$$

where $s_k>0$ is the step size (which, given the notation, possibly depends on the current iteration index k), $\beta^{-1}>0$ stands for the intensity of isotropic Gaussian noise $\mathcal{N}(\boldsymbol{0},\boldsymbol{I}_D)$ (often referred to as a temperature) and the term $-2\lambda s_k\boldsymbol{\theta}_i^k$ corresponds to the regularization in (2.25).[3]

In these terms, one of the main results of mean-field theory for NNs [MMN18] states that along the trajectory of SGD (2.30) the network weights remain independent as in (2.28). In statistical physics (in particular, in topics related to large systems of interacting particles) such a phenomenon is often referred to as *propagation of chaos* (see, e.g., [Szn91]). In particular, for some $\varepsilon>0$, we assume that the step size of the noisy SGD update (2.30) is given by $s_k=\varepsilon\cdot\xi(\varepsilon k)$, where $\xi:\mathbb{R}_{\geq 0}\to\mathbb{R}_{\geq 0}$ is a sufficiently regular function, meaning:

$$t\mapsto\xi(t)\text{ is bounded Lipschitz, i.e., }\|\xi\|_\infty,\|\xi\|_{\mathrm{Lip}}\leq C,\text{ with }\int_0^{+\infty}\xi(t)\mathrm{d}t=\infty. \tag{2.31}$$

for some universal constant $C>0$. Condition (2.31), as a special case, includes the constant step size, meaning that $\xi(t)=\text{const}$. Let

$$\hat{\rho}_k^N:=\frac{1}{N}\sum_{i=1}^{N}\delta_{\boldsymbol{\theta}_i^k}$$

denote the empirical distribution of weights after k steps of noisy SGD. Then, in [MMN18], it is proved that the evolution of the empirical distribution of the network weights $\hat{\rho}_k^N$ is well approximated by a certain distributional dynamics ρ_t. In formulas,

$$\hat{\rho}_{\lfloor t/\varepsilon\rfloor}^N\rightharpoonup\rho_t \tag{2.32}$$

almost surely along any sequence $(N\to\infty,\varepsilon_N\to 0)$ such that $N/\log(N/\varepsilon_N)\to\infty$ and $\varepsilon_N\log(N/\varepsilon_N)\to 0$. Here, we have put the subscript N in ε_N to emphasize that the choice of the learning rate depends on N, $\mu\rightharpoonup\nu$ denotes weak L_1 convergence of measures and $\lfloor a\rfloor$ defines the closest integer that is not greater than $a\in\mathbb{R}$. Moreover, ρ_t solves the following partial differential equation (PDE)

$$\begin{aligned}\partial_t\rho_t&=2\xi(t)\nabla_{\boldsymbol{\theta}}\cdot(\rho_t\nabla_{\boldsymbol{\theta}}\Psi_\lambda(\boldsymbol{\theta},\rho_t))+2\xi(t)\beta^{-1}\Delta_{\boldsymbol{\theta}}\rho_t,\\ \Psi_\lambda(\boldsymbol{\theta},\rho):&=\mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}}\left[\left(y_\rho^{\sigma^*}(\boldsymbol{x})-y\right)\cdot\sigma^*(x,\boldsymbol{\theta})\right]+\frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2,\end{aligned} \tag{2.33}$$

[3] Let us point out right away that the additional regularization (2.25) and Gaussian noise in (2.30) ensures the necessary regularity required for the analysis. However, they are not important for the conceptual understanding of the mean-field approach.

where $y_{\rho_t}^{\sigma^*}$ is defined analogously to $y_{\rho_0}^{\sigma^*}$ in (2.28), i.e., define "infinite-width" network with activation σ^* and weight distribution $\rho : \mathbb{R}^D \to [0, +\infty)$ as follows:

$$y_\rho^{\sigma^*}(\boldsymbol{x}) = \int \sigma^*(\boldsymbol{x}, \boldsymbol{\theta}) \rho(\mathrm{d}\boldsymbol{\theta}),$$

$\nabla_{\boldsymbol{\theta}} \cdot \boldsymbol{v}(\boldsymbol{\theta})$ denotes the divergence of the vector field $\boldsymbol{v}(\boldsymbol{\theta})$ and $\Delta_{\boldsymbol{\theta}}$ stands for the Laplace operator, i.e., $\Delta_{\boldsymbol{\theta}} g(\boldsymbol{\theta}) := \sum_{j=1}^D \partial_{\theta_j}^2 g(\boldsymbol{\theta})$ for $g : \mathbb{R}^D \to \mathbb{R}$.

The PDE in (2.33) is often referred to as the *continuity equation*. We now briefly elaborate why. Another way to rewrite (2.33) is as follows, for some vector field $\boldsymbol{v}(\boldsymbol{\theta})$:

$$\frac{\partial}{\partial t} \rho_t = -\nabla \cdot (\rho_t \boldsymbol{v}(\boldsymbol{\theta})). \tag{2.34}$$

For a regular enough region of space $\Omega \subset \mathbb{R}^D$ and its enclosing area $\partial\Omega$ (boundary), the *law of mass conservation* can then be written as follows:

$$\frac{\partial}{\partial t} \int_\Omega \rho_t(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} = -\int_{\delta\Omega} \rho_t \langle \boldsymbol{v}, \boldsymbol{n} \rangle \mathrm{d}\boldsymbol{\sigma}, \tag{2.35}$$

where $\mathrm{d}\boldsymbol{\sigma}$ stands for an "infinitesimal" area element and $\boldsymbol{n}$ denotes the normal vector to the boundary $\delta\Omega$ at the point $\boldsymbol{\theta} \in \partial\Omega$. In words, the LHS of (2.35) describes the change of mass (the number of particles) inside the volume Ω, while the RHS of (2.35) defines how many particles escape Ω through the boundary $\partial\Omega$ under the influence of the vector field $\boldsymbol{v}(\boldsymbol{\theta})$. In these terms, (2.35) acts as a consistency equation: the particles do not magically "teleport" from the volume without passing the boundary. We now outline why (2.35) is equivalent to (2.34). This follows from an application of the *divergence theorem*, namely, integrating (2.34) over Ω gives (under suitable regularity):

$$\frac{\partial}{\partial t} \int_\Omega \rho_t(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} = -\int_\Omega \nabla \cdot (\rho_t \boldsymbol{v}(\boldsymbol{\theta})) \mathrm{d}\boldsymbol{\theta} = [\text{Divergence Theorem}] = -\int_{\delta\Omega} \rho_t \langle \boldsymbol{v}, \boldsymbol{n} \rangle \mathrm{d}\boldsymbol{\sigma}.$$

For the forthcoming analysis, a certain regularity is required for the weight distribution ρ. In particular, the weight distribution is restricted to a set of admissible densities

$$\mathcal{K} := \left\{ \rho : \mathbb{R}^D \to [0, +\infty) \text{ measurable: } \int \rho(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} = 1,\ M(\rho) < \infty \right\},$$

where

$$M(\rho) := \int \|\boldsymbol{\theta}\|_2^2 \rho(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}$$

stands for the second moment of distribution ρ. The expected risk attained on the distribution ρ by the infinite-width network with activation σ^* is defined by

$$R^{\sigma^*}(\rho) := \mathbb{E}_{(\boldsymbol{x}, y) \sim \mathbb{P}} \left[\left(y_\rho^{\sigma^*}(\boldsymbol{x}) - y \right)^2 \right].$$

The quantity

$$H(\rho) := -\int \rho(\boldsymbol{\theta}) \log \rho(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}$$

stands for the differential entropy of ρ, which is equal to $-\infty$ if the distribution ρ is singular. In this view, the distributional dynamics (2.33) minimizes the free energy

$$\mathcal{F}^{\sigma^*}(\rho) = \frac{1}{2} \cdot R^{\sigma^*}(\rho) + \frac{\lambda}{2} \cdot M(\rho) - \beta^{-1} H(\rho), \tag{2.36}$$

over the set of admissible densities $\mathcal{K}$. Furthermore, this free energy has a unique minimizer and the solution of (2.33) converges to it as $t \to \infty$:

$$\rho_t \rightharpoonup \rho^*_{\sigma^*} \in \arg\min_{\rho' \in \mathcal{K}} \mathcal{F}^{\sigma^*}(\rho'), \qquad \text{as } t \to \infty.$$

Given the previous discussion on the continuity equation (2.34), we can now define the related vector field $\boldsymbol{v}(\boldsymbol{\theta})$ more formally. Since the evolution in (2.33) minimizes the free energy, it is quite natural to expect the dynamics to be governed by a gradient descent on the functional $\mathcal{F}^{\sigma^*}$. This concept is formalized by the *first variation* of the functional $\mathcal{F}^{\sigma^*}$. The first variation $\frac{\partial \mathcal{F}^{\sigma^*}}{\partial \rho} : \mathcal{P}(\mathbb{R}^D) \times \mathbb{R}^D \to \mathbb{R}$ is defined as a continuous functional such as

$$\lim_{\epsilon \to 0} \frac{\mathcal{F}^{\sigma^*}(\varepsilon\nu + (1-\varepsilon)\rho) - \mathcal{F}^{\sigma^*}(\rho)}{\epsilon} = \int \frac{\partial \mathcal{F}^{\sigma^*}(\rho)}{\partial \rho}(\boldsymbol{\theta}) \mathrm{d}(\nu - \rho)(\boldsymbol{\theta})$$

where $\nu \in \mathcal{P}(\mathbb{R}^D)$ is an arbitrary perturbation. In these terms, the vector field $\boldsymbol{v}(\boldsymbol{\theta})$ in the continuity equation (2.34) is simply the gradient $\nabla_{\boldsymbol{\theta}}$ of the first variation of the free energy functional.

A few remarks are in order regarding the gradient flow in (2.33). Recall that the Wassertein-2 distance is defined as follows:

$$W_2(\mu, \nu) := \left(\inf_{\gamma \in \mathcal{C}(\mu,\nu)} \int \left\| \boldsymbol{\theta} - \hat{\boldsymbol{\theta}} \right\|_2^2 \gamma(\mathrm{d}\boldsymbol{\theta}, \mathrm{d}\hat{\boldsymbol{\theta}}) \right)^{1/2},$$

where $\mathcal{C}(\mu, \nu)$ denotes the space of couplings, i.e., measures on the product space $\mathbb{R}^D \times \mathbb{R}^D$ the marginals of which coincide with μ and ν respectively. In this view, the PDE in (2.33) is viewed as a gradient flow in the space of probability measures endowed with W_2 metric. Namely, heuristically speaking the fact that ρ_t solves (2.33) is equivalent to, for some small $t' > 0$:

$$\rho_{t+t'} = \arg\min_{\rho \in \mathcal{P}(\mathbb{R}^D)} \left\{ \mathcal{F}^{\sigma^*}(\rho) + \frac{1}{2\xi(t)t'} \cdot W_2(\rho, \rho_t) \right\} \tag{2.37}$$

The variational problem in (2.37) is referred as the JKO scheme, which stands for the first letter of authors' names in [JKO98], and, in a nutshell, corresponds to a W_2 equivalent of Euler's discretization for a gradient flow in $\mathbb{R}^D$.

Regarding the minimizer of the free energy (2.36), one could notice that given an integral form of $y^{\sigma^*}_\rho$ the functional $\mathcal{F}(\rho)$ is convex in ρ. In fact, it is strongly convex in ρ as the differential entropy $H(\rho)$ is a strongly convex functional. In this view, the minimizer $\rho^*_{\sigma^*}$ of the free energy (2.36) is unique. Moreover, it has a "pleasant" functional representation often referred to as Gibbs form:

$$\rho^*_{\sigma^*}(\boldsymbol{\theta}) = \frac{1}{Z_{\sigma^*}(\beta, \lambda)} \cdot \exp\{-\beta \Psi_\lambda(\boldsymbol{\theta}, \rho^*_{\sigma^*})\}, \tag{2.38}$$

where $Z_{\sigma^*}(\beta, \lambda)$ stands for the partition function, which ensures that $\rho^*_{\sigma^*}$ is a valid probability distribution, i.e., $\int \rho^*_{\sigma^*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} = 1$. A non-rigorous way to see that the functional form (2.38) corresponds to the correct candidate is to treat distribution the ρ in (2.36) as a "number" and take the derivative of the corresponding Lagrangian $\mathcal{L}(\rho)$, i.e.,

$$\frac{\partial}{\partial \rho}[\mathcal{L}(\rho)] = \frac{\partial}{\partial \rho}\left[\mathcal{F}(\rho) - \gamma\left(\int \rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} - 1\right)\right] = \Psi_\lambda(\boldsymbol{\theta}, \rho) + \beta \log \rho(\boldsymbol{\theta}) + \beta - \gamma \tag{2.39}$$

where the $\gamma \in \mathbb{R}$ term corresponds to the term of the Lagrangian which enforces the constraint $\int \rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} = 1$. In this view, from (2.39) it is easy to see that $\rho^*_{\sigma^*}$ indeed has the form described in (2.38):

$$\frac{\partial}{\partial \rho}\left[\mathcal{L}(\rho)\right] = 0 \Rightarrow \rho^*_{\sigma^*}(\boldsymbol{\theta}) \propto \exp\{-\beta\Psi_\lambda(\boldsymbol{\theta}, \rho^*_{\sigma^*})\}.$$

Another way to assert that the minimizer of the free energy (2.36) has the form of Gibbs distribution is to invoke the "maximum entropy" principle (see, for instance, [GMZ09]). Namely, the free energy (2.36) is composed of three terms: the approximation error, the second moment and the entropy of ρ. In this view, fixing the first two terms, which correspond to the expected value of statistics of the related exponential family distribution, and finding the maximum entropy distribution given this constraint will exactly yield the form in (2.38).

2.3 Autoencoders and Related Concepts

Lossy compression and autoencoders. We start this section by covering a few fundamentals of the *lossy compression*. Consider the problem of obtaining a lower-dimensional (compressed) representation of a sequence $\boldsymbol{x} \in \mathbb{R}^d$ of length d. Namely, we aim to find an "encoding" mapping $E : \mathbb{R}^d \to \mathbb{R}^n$ such that the related compressed representation $\boldsymbol{z}$, i.e.,

$$\boldsymbol{z} = E(\boldsymbol{x}) \in \mathbb{R}^n,$$

is more length-efficient and retains as much information about the initial $\boldsymbol{x}$ as possible, given the rate constraint:

$$r = \frac{n}{d} = \text{const}.$$

Namely, for each encoding scheme $E(\cdot)$ one can associate the "decoding" procedure $D : \mathbb{R}^n \to \mathbb{R}^d$. In this view, the performance of the compression scheme is then evaluated based on the *distortion* measure achieved for a fixed rate r:

$$\text{distortion}(r) = \mathbb{E}\left[\text{dist}(\boldsymbol{x}, D(\boldsymbol{z}))\right],$$

where the expectation is taken of the distribution of the source signal $\boldsymbol{x}$. The distance measure $\text{dist} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ between the input $\boldsymbol{x}$ and its reconstruction $\hat{\boldsymbol{x}} = D(\boldsymbol{z}) \in \mathbb{R}^d$ is usually picked based on the nature of the source data itself. For instance, for binary inputs $\boldsymbol{x} \in \{0,1\}^d$ it is quite natural to consider the *Hamming* distance, i.e.,

$$\text{dist}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{d}\sum_{i=1}^{d} \mathbb{1}\{x_i \neq \hat{x}_i\},$$

while for a real-valued source $\boldsymbol{x} \in \mathbb{R}^d$ the common choice is usually squared distortion:

$$\text{dist}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{d} \cdot \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2.$$

Given the above discussion a natural question arises: what is the best possible compression performance one could hope to achieve? This question is addressed via the concept of the *rate-distortion* function (see, for example, [CT06]). The rate-distortion function corresponds to the following variational problem over encoding-decoding pair:

$$\begin{aligned} &\inf_{E,D \in \mathcal{Q}} r, \\ &\mathcal{Q} := \{(E,D) : \mathbb{E}\left[\text{dist}(\boldsymbol{x}, D(\boldsymbol{z}))\right] \leq \hat{D}\}. \end{aligned} \tag{2.40}$$

In words, (2.40) corresponds to finding the smallest rate r such that there exists an encoding-decoding pair for which the associated distortion is not exceeding the critical value $\hat{D}$.

While the variational problem formulation in (2.40) is quite intuitive, solving it, especially for finite d, is a challenging task. Luckily, rate-distortion admits another equivalent (under suitable assumptions) information-theoretic formulation in the asymptotic limit $d \to \infty$ (infinite sequence length) for an i.i.d. sequence, i.e., $x_i \overset{\text{i.i.d.}}{\sim} p_X(x)$. Namely, the following holds

$$\begin{aligned} \lim_{d \to \infty} r(\hat{D}) &= \inf_{p_{\hat{X}|X}(\hat{x}|x) \in \mathcal{Q}} I(\hat{X}, X), \\ \mathcal{Q} &:= \{p_{\hat{X}|X}(\hat{x}|x) : \mathbb{E}_{p_{\hat{X}|X}(\hat{x}|x) p_X(x)} [\text{dist}(\hat{x}, x)] \leq \hat{D}\}, \end{aligned} \tag{2.41}$$

where $I(\hat{X}, X)$ stands for the *mutual information* between the source signal $X \in \mathbb{R} \sim p_X(x)$ and its reconstruction $\hat{X} \in \mathbb{R}$, i.e.,

$$I(\hat{X}, X) = H(\hat{X}) - H(\hat{X}|X), \tag{2.42}$$

here $H(\hat{X})$ denotes the entropy of the distribution of the reconstruction and $H(\hat{X}|X)$ is the conditional entropy of the reconstruction given the corresponding input X. In formulas,

$$\begin{aligned} H(\hat{X}) &:= -\int p_{\hat{X}}(\hat{x}) \cdot \log_2 p_{\hat{X}}(\hat{x}) \mathrm{d}\hat{x}, \\ H(\hat{X}|X) &:= -\int p_{\hat{X}|X}(\hat{x}|x) p_X(x) \cdot \log_2 p_{\hat{X}|X}(\hat{x}|x) \mathrm{d}\hat{x} \mathrm{d}x, \end{aligned} \tag{2.43}$$

In this formulation, the conditional $p_{\hat{X}|X}(\hat{x}|x)$ is sometimes referred as a *test channel*.

The advantage of (2.41) over (2.40) (modulo, i.i.d. assumption) is that the mutual information could be lower-bounded using the RHS of (2.42), and the tightness of the bound could be assured by constructing a suitable test channel. For instance, using this approach one could compute the rate-distortion of two common signal instances: for a $\text{Bernoulli}(p)$ signal with *Hamming distortion*

$$X \sim p \cdot \delta_1 + (1 - p) \cdot \delta_0$$

we obtain

$$r(\hat{D}) = \begin{cases} H_b(p) - H_b(\hat{D}), & 0 \leq \hat{D} \leq \min\{p, 1 - p\}, \\ 0, & \hat{D} > \min\{p, 1 - p\}, \end{cases} \tag{2.44}$$

where $H_b(a)$ for $a \in (0, 1)$ stands for the binary entropy, i.e.,

$$H_b(a) := -a \cdot \log_2 a - (1 - a) \cdot \log_2(1 - a);$$

for Gaussian signal $X \sim \mathcal{N}(0, \sigma^2)$ with *squared distortion* we get

$$r(\hat{D}) = \begin{cases} \frac{1}{2} \log_2 \left(\frac{\sigma^2}{\hat{D}} \right), & 0 \leq \hat{D} \leq \sigma^2, \\ 0, & \hat{D} > \sigma^2, \end{cases} \tag{2.45}$$

Specifically, for a $\text{Bernoulli}(1/2)$ signal the rate-distortion yields a natural result, since such signal has 1-bit of information. Thus, we would expect that for the rates $r \geq 1$ the corresponding compression is *lossless*, i.e., the associated distortion is vanishing. Compression of a Gaussian signal is never lossless for finite rate r since one would not expect to be able to represent a *real*-valued signal using a discrete form without any precision drawbacks.

While the rate-distortion function and its particular expression in (2.42) is a highly useful concept, its closed-form expression is usually unavailable even for sources which are just slightly more complicated than described above. For instance, there is no closed form expression akin to (2.44) and (2.45) available for *sparse* Gaussian data

$$x \sim p \cdot \mathcal{N}(0, \sigma^2) + (1-p) \cdot \delta_0.$$

However, at least for scalar signals (i.e., $\mathrm{i.i.d.}$ sequence) one could access the optimal mutual information value in (2.41) via numerical simulation, for instance, using Blahut-Arimoto algorithm [Bla72, Ari72]. Nevertheless, for high-dimensional correlated signals, which are a common practical scenario, currently there is no known way to evaluate reliably the rate-distortion (2.40). However, it is important to mention that a considerable effort was made towards accessing an approximate value of the rate-distortion for natural signals, see, for example, [YM21a, LHB22].

Given a general rise of deep learning in the recent years, it is no surprise that there was a substantial effort to adopt neural architectures for compression purposes. A particular paradigm of *auto-encoding* stands out as one of the "go-to" options. The high-level idea of auto-encoding is quite simple - instead of using a predefined scheme for compression, parameterize an encoding-decoding pair by a suitable neural network and train the whole system via back-propagation to minimize the reconstruction error. Of particular interest to the current thesis are the so called "shallow" autoencoders, i.e.,

$$\hat{\boldsymbol{x}}(\boldsymbol{x}) = \boldsymbol{A} \cdot \sigma(\boldsymbol{B} \cdot \boldsymbol{x}), \quad \boldsymbol{A} \in \mathbb{R}^{d \times n}, \quad \boldsymbol{B} \in \mathbb{R}^{n \times d}, \tag{2.46}$$

for some scalar processing function $\sigma : \mathbb{R} \to \mathbb{R}$. In words, for a model (2.46) the encoding corresponds to a linear transformation of the initial $\boldsymbol{x}$ and a subsequent elementwise application of the non-linearity $\sigma(\cdot)$:

$$\boldsymbol{z} = \sigma(\boldsymbol{B} \cdot \boldsymbol{x}) \in \mathbb{R}^n. \tag{2.47}$$

The decoding in this case is also quite simple: the compressed representation $\boldsymbol{z}$ of the input $\boldsymbol{x}$ is linearly mapped back to the initial space with matrix multiplication by $\boldsymbol{A}$.

Despite the simplicity of a shallow model (2.46) there are not many results available for the case of a non-linear processing function, while the linear case ($\sigma(x) \propto x$) exhibits a PCA-like behaviour [KBGS19]. In case of a non-linearity, the analysis is limited to the "extreme" regimes. Namely, the model is either severely under-parameterized [RG22, CZ23], resulting in a vanishing rate $r \to 0$, or highly over-parameterized [Ngu21] ($r \to \infty$). In this thesis, we focus on the challenging *proportional* regime, for which $r = n/d = \mathrm{const}$ (while sometimes allowing $d \to \infty$ but keeping the ratio fixed), in the context of a *one-bit compression* problem, which means that a non-linearity $\sigma(\cdot)$ is fixed to the $\mathrm{sign}(\cdot)$ activation.

Approximate message passing. A shallow encoding scheme described in (2.47) is often referred to in the statistical literature as a *generalized linear model* (GLM). This particular design instance was widely studied in the context of signal recovery. Namely, we are interested in recovering the signal $\boldsymbol{x} \in \mathbb{R}^d$ given n of its non-linear projected observations

$$\boldsymbol{z} = \sigma(\boldsymbol{B} \cdot \boldsymbol{x}) \in \mathbb{R}^n.$$

Notice that the signal recovery problem in this formulation bares striking resemblance to the decoding procedure. In fact, if the encoding structure is fixed, the problem of recovery is equivalent to finding a suitable decoding scheme.

Before delving into the specifics of the GLM design (2.47), it is useful to briefly introduce the framework of Bayesian inference (see, for example, [Mur22] for concise but more elaborate introduction), under which the problems akin to (2.47) are analyzed. The Bayesian statistician starts with the distributional assumption on the input signal $\boldsymbol{x}$, that is referred to as a *prior* distribution $p_{\boldsymbol{X}}(\boldsymbol{x})$. A particular choice of prior distribution is usually tailored to the nature of the problem. For the purposes of the current thesis, the prior distribution is given, since we are focused on the compression of a known signal. However, Bayesian framework (at least on a surface level) is quite flexible in imposing the structural assumptions on the inputs by choosing the prior distribution accordingly, even when the true signal nature is not available. To illustrate this, consider the weaker version of (2.47), namely recovering $\boldsymbol{x}$ given the set of its linear observations:

$$\boldsymbol{z} = \boldsymbol{B}\boldsymbol{x} \in \mathbb{R}^n. \tag{2.48}$$

Assume that we are interested not just in some $\boldsymbol{x}$ but in its "most sparse" version. In this view, we could formulate the recovery problem as follows (for some sensitivity $\lambda > 0$):

$$\arg\min_{\boldsymbol{x}} \left\{ \frac{1}{2} \cdot \|\boldsymbol{z} - \boldsymbol{B}\boldsymbol{x}\|_2^2 + \lambda \cdot \|\boldsymbol{x}\|_1 \right\}, \tag{2.49}$$

which is famously known as the LASSO regression (see, for instance, [BBV04]). The choice of specific L_1 penalty enforces the desired sparsity constraint. While (2.49) makes a lot of sense given the formulation of the problem, how come it is connected to the Bayesian approach? For this particular equivalence, assume that the prior distribution follows Laplace law, i.e.,

$$p_{\boldsymbol{X}}(\boldsymbol{x}) \propto \exp\left(-\lambda \cdot \|\boldsymbol{x}\|_1\right),$$

and consider *Gaussian likelihood* model:

$$p(\boldsymbol{z}|\boldsymbol{B}, \boldsymbol{x}) \propto \exp\left(-\frac{1}{2}\|\boldsymbol{z} - \boldsymbol{B}\boldsymbol{x}\|_2^2\right).$$

Given this, Bayesian statistician will aim to access the posterior distribution of the signal

$$p(\boldsymbol{x}|\boldsymbol{B}, \boldsymbol{z}) \propto p(\boldsymbol{z}|\boldsymbol{B}, \boldsymbol{x}) p_{\boldsymbol{X}}(\boldsymbol{x}). \tag{2.50}$$

In this view, picking the mode of (2.50), which is often referred to as *maximum a posteriori (MAP)* estimate, will exactly correspond to the LASSO objective (2.49). However, the MAP estimate is usually sub-optimal as it does not utilize fully the posterior distribution (2.50). In case of the squared reconstruction error, that appears in the LASSO formulation (2.49), the Bayes optimal predictor corresponds to the conditional expectation

$$\mathbb{E}[\boldsymbol{x}|\boldsymbol{B}, \boldsymbol{z}] = \int \boldsymbol{x} \cdot p(\boldsymbol{x}|\boldsymbol{B}, \boldsymbol{z}) \mathrm{d}\boldsymbol{x}, \tag{2.51}$$

which gives the optimal value of the corresponding mean squared error:

$$\arg\min_{f} \mathbb{E}_{p_{\boldsymbol{X}}(\boldsymbol{x})} \left[\|\boldsymbol{x} - f(\boldsymbol{z})\|_2^2\right], \tag{2.52}$$

where $f : \mathbb{R}^n \to \mathbb{R}^d$. However, as usual, the devil is in the details. Namely, the computation of the conditional $\mathbb{E}[\boldsymbol{x}|\boldsymbol{B}, \boldsymbol{z}]$ requires knowledge of the normalization constant in (2.50). While it is possible to approach the problem via generic message passing algorithms [YFW+03], such approach is prone to scale poorly in the system size (dimensions of $\boldsymbol{B}$) and does not allow for a precise theoretical analysis in most cases (the corresponding factor graph structure is not "tree"-like).

The salvation lies in the high-dimensional regime and a *random* design of the matrix $\boldsymbol{B}$. In a nutshell, in the high-dimensional limit $d \to \infty$ for the random $\boldsymbol{B}$ (such as Gaussian or orthogonal ensemble, more on the latter will be described further) the message passing algorithm simplifies due to the concentration of measure phenomenon. This gives rise to the family of *approximate* message passing (AMP) algorithms. AMP is a class of iterative algorithms that was successfully applied in a number of statistical inference problems such as linear regression [DMM09, BM11], low-rank matrix estimation [MT13, FR18, MV21], and, of a particular interest to the current thesis, generalized linear models [SRF16, MV22, VKM22]. One of the key advantages of using AMP algorithm is so called state evolution equations. Informally, the state evolution provides the characterization of each of the AMP iterates in the high-dimensional limit $d \to \infty$. This characterization provides the necessary tools to analyze the AMP iterates, which out-of-the-box message passing algorithm (e.g., "loopy" belief propagation) is missing. Moreover, via the "replica" ansatz from statistical mechanics [TCVS13] the state evolution equations may be viewed as a fixed point iteration performed to optimize the *free energy* (for the GLM case see [BKM$^+$19]). Quite conveniently, the free energy may be linked via "I-MMSE" argument (cf. Corollary 4 in [BKM$^+$19]) to the Bayes optimal MSE (2.52). This link implies that in the high-dimensional limit suitable AMP algorithm will saturate the conjectured optimal performance, which in turn means that if the corresponding assumptions hold true approximate message passing is the optimal algorithm for the job.

Moving to the case of the general linear model (2.47), we will focus on the particular instance of AMP algorithm for a bi-rotationally invariant sensing design of $\boldsymbol{B}$ - rotationally invariant generalized AMP (RI-GAMP) [VKM22]. We first describe the specific choice of random ensemble for the design matrix $\boldsymbol{B}$. The encoding matrix $\boldsymbol{B}$ is *stochastic* and restricted to the bi-rotationally invariant family:

$$\boldsymbol{B} = \boldsymbol{O}^\top \boldsymbol{\Lambda} \boldsymbol{Q}, \tag{2.53}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{n \times d}$ is a diagonal matrix of the singular values of $\boldsymbol{B}$, $\boldsymbol{O}$ and $\boldsymbol{Q}$ are independent Haar matrices ($\mathrm{i.i.d.}$ samples from special orthogonal group). The important assumption here, for further AMP analysis, is that the spectral distribution of $\boldsymbol{B}$ is well behaved in the high-dimensional limit $d \to \infty$ with $r = n/d = \mathrm{const}$. This means that the empirical distribution of the singular values converges (in a certain sense) to the fixed spectral distribution ρ_Λ:

$$\frac{1}{n}\sum_{i=1}^{n} \delta_{\Lambda_i} \to \rho_\Lambda, \quad d \to \infty.$$

With these technicalities out of the way we are ready to state the RI-GAMP algorithm. In essence, the AMP algorithm consists of consecutive iterates that "ping-pong" between the signal estimates $\hat{\boldsymbol{x}}^t$ and the observation refinements $\hat{\boldsymbol{z}}^t$ as follows

$$\begin{aligned} \boldsymbol{x}^t &= \boldsymbol{B}^\top \cdot \hat{\boldsymbol{z}}^t - \sum_{i=1}^{t-1} \beta_{t,i} \cdot \hat{\boldsymbol{x}}^i, \quad \hat{\boldsymbol{x}}^t = f_t(\boldsymbol{x}^1, \cdots, \boldsymbol{x}^t), \\ \boldsymbol{z}^t &= \boldsymbol{B} \cdot \hat{\boldsymbol{x}}^t - \sum_{i=1}^{t} \alpha_{t,i} \cdot \hat{\boldsymbol{z}}^i, \quad \hat{\boldsymbol{z}}^{t+1} = g_t(\boldsymbol{z}^1, \cdots, \boldsymbol{z}^t, \hat{\boldsymbol{z}}^1), \end{aligned} \tag{2.54}$$

for some choice of *denoising* functions f_t and g_t which are applied "row-wise", i.e.,

$$\hat{x}_i^t = f_t(x_i^1, \cdots, x_i^t), \quad i \in \{1, \cdots, d\},$$

and $\hat{\boldsymbol{z}}^1 = \boldsymbol{z}$. The terms containing $\beta_{t,i}$ and $\alpha_{t,i}$ are referred as Onsager corrections. In particular, the correction terms ensure that the aforementioned state-evolution characterization of the iterates $\hat{\boldsymbol{x}}^t$ holds in the high-dimensional limit $d \to \infty$.

For the purpose of the upcoming discussion we assume that $\boldsymbol{x}$ has $\mathrm{i.i.d.}$ components distributed according to p_X, which is the setting of interest for the current thesis application of the AMP algorithm. In these terms, the state-evolution characterization allows to access the "statistics" of iterates in (2.54), in particular, $\boldsymbol{x}^t$, via its 1-dimensional counterpart. More formally, let $\psi : \mathbb{R}^t \to \mathbb{R}$ be a pseudo-Lipschitz function of order 2, i.e.,

$$|\psi(\boldsymbol{a}) - \psi(\boldsymbol{b})| \ \leq L \cdot \|\boldsymbol{a} - \boldsymbol{b}\|_2 \cdot (1 + \|\boldsymbol{a}\|_2 + \|\boldsymbol{b}\|_2), \quad \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^t,$$

for some constant $L > 0$, then the following holds:

$$\lim_{d \to \infty} \frac{1}{d} \sum_{i=1}^{d} \psi(x_i^1, \cdots, x_i^t) = \mathbb{E}\left[\psi(X_1, \cdots, X_t)\right], \tag{2.55}$$

where the vector $\boldsymbol{X} = (X_1, \cdots, X_t)$ has the following joint law

$$\boldsymbol{X} = \boldsymbol{\mu}_t \cdot X + \boldsymbol{g}, \quad X \sim p_X, \quad \boldsymbol{g} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_t)$$

where $\boldsymbol{g}$ is independent of X and the state-evolution parameters $\boldsymbol{\mu}_t \in \mathbb{R}^t$ and $\boldsymbol{\Sigma}_t \in \mathbb{R}^{t \times t}$ can be computed analytically given the choice of denoisers $f_t(\cdot)$ and $g_t(\cdot)$. In simpler terms, (2.55) states that for all intents and purposes one could assume that at each iteration $\boldsymbol{x}^t$ is a mixture of the initial signal $\boldsymbol{x}$ and some independent Gaussian noise. In this view, the fact that $f_t(\cdot)$ is referred to as denoising function makes a lot of sense, since it tries to "denoise" a noisy Gaussian observation of $\boldsymbol{x}$. Let us remark that the evaluation of a "statistic" in (2.55) is particularly useful to access the reconstruction error for an autoencoder model analyzed in Chapter 6, as the squared error is a pseudo-Lipschitz function of order 2.

We also note that RI-GAMP algorithm has quite a few "degrees of freedom" by its design. Namely, we are free to choose the denoising functions f_t and g_t and, to achieve the optimal "encoding", the spectral distribution of the ensemble $\boldsymbol{\Lambda}$. However, such flexibility raises a natural question: which choice is the best given the problem at hand. The answer might seem a bit unsatisfactory in this case - it ultimately depends. We now illustrate this with two particular design examples. Concerning the choice of ρ_Λ, the work [MXM21] provides the analysis of the *expectation propagation* [Min01] algorithm, that is closely related to the approximate message passing. Namely, the authors characterize the optimal (MSE-wise) spectral distribution of $\boldsymbol{\Lambda}$ given the choice of the activation function σ in the GLM design (2.47). Regarding the denoising, for Gaussian design of $\boldsymbol{B}$ and Gaussian inputs $\boldsymbol{x}$ the denoising functions in (2.54) simplify as it is optimal to condition only on the immediate past instead of the whole "history" [FVR+22]. However, for general designs and priors, it is unclear which choice is optimal. In Chapter 6, we empirically show that using only a subset of the past and a few AMP iterations could lead to nearly Bayes optimal performance for a specific case. Nevertheless, it is far from being a universal recipe and one should empirically validate on the case by case basis which procedure yields the best performance.

Hermite polynomials. We now briefly describe the particular series expansion that comes in handy for the analysis of one-bit compression of a Gaussian source in Chapters 5 and 6. The "probabilitic" Hermite polynomial maybe defined via so called Rodrigues' formula (see, for instance, [Rad08]):

$$H_n(x) = (-1)^n \cdot e^{\frac{x^2}{2}} \cdot \left(\frac{d}{dx}\right)^n \left[e^{-\frac{x^2}{2}}\right] = (-1)^n \cdot e^{\frac{x^2}{2}} \cdot D^n \left[e^{-\frac{x^2}{2}}\right]. \tag{2.56}$$

It is evident from (2.56) that H_m are indeed polynomials.

Using Rodrigues's formula it is quite easy to show that $\{H_n\}_{n=0}^{\infty}$ forms an orthogonal set with respect to the standard Gaussian measure μ. Indeed, using (2.56) we can write for $m < n$

$$\int H_m(x)H_n(x)e^{-\frac{x^2}{2}}\mathrm{d}x = (-1)^n \int H_m(x)D^n\left[e^{-\frac{x^2}{2}}\right]\mathrm{d}x.$$

At this point, we can integrate by parts n times and push the derivative every-time in H_m. Note that every time the term which corresponds to the "boundary" will vanish due to the exponential decay of $e^{-\frac{x^2}{2}}$ at infinity. Combining these observations gives

$$\int H_m(x)H_n(x)e^{-\frac{x^2}{2}}\mathrm{d}x = \frac{(-1)^{n+m}}{2^n} \cdot \int D^n\left[H_m(x)\right]e^{-\frac{x^2}{2}}\mathrm{d}x, \tag{2.57}$$

which is vanishing since $m < n$. The RHS of (2.57) also suggests a way to rescale H_n to make $\{H_n\}_{n=0}^{\infty}$ an *orthonormal* set. From here on, with an abuse of notation, we will use the normalized version of Hermite polynomials.

A slightly less trivial observation corresponds to the fact that Hermite polynomials are dense in $L^2(\mathbb{R}, \mu)$, i.e., space of L^2 integrable functions with respect to standard Gaussian measure. This in conjunction with the fact that $\{H_n\}_{n=0}^{\infty}$ is an orthonormal set implies that the Hermite polynomials are a suitable basis in $L^2(\mathbb{R}, \mu)$. Formally, for any $f \in L^2(\mathbb{R}, \mu)$ the following Hermite expansion holds

$$f(x) = \sum_{\ell=0}^{\infty} c_\ell \cdot H_\ell(x). \tag{2.58}$$

The expansion (2.58) is very useful for the upcoming analysis in Chapters 5 and 6 since it allows to obtain a closed form expression of the reconstruction error for a shallow autoencoder model (2.46). The corresponding computation relies on the *reproducing* property of Hermite polynomials. The reproducing property implies the following:

$$\mathbb{E}_{x,y}\left[H_n(x)H_m(y)\right] = \rho^n \cdot \delta_{n,m} \tag{2.59}$$

for two ρ-correlated Gaussians for $\rho \in [-1, 1]$, i.e., $x, y \sim \mathcal{N}(0, 1)$ and $\mathbb{E}[xy] = \rho$, where $\delta_{n,m} = 0$ if $n \neq m$ and 1 otherwise. This allows to conclude that

$$\mathbb{E}_{x,y}\left[f(x)f(y)\right] = \sum_{\ell=0}^{\infty} c_\ell^2 \cdot \rho^\ell. \tag{2.60}$$

For the particular case of $f \equiv \mathrm{sign}$ activation, the following Grothendieck's identity (see, e.g., Lemma 3.6.6 in [Ver18]) holds:

$$\mathbb{E}\left[f(x) \cdot f(y)\right] = \frac{2}{\pi} \cdot \arcsin(\rho).$$

To show that (2.59) holds, it is useful to note that distributionally (x, y) and

$$(x, \rho \cdot x + \sqrt{1 - \rho^2} \cdot g), \quad g \perp x, \quad g \sim \mathcal{N}(0, 1)$$

are equivalent. Now in order to proceed with (2.59) it is useful to study the generating function of the pair (x, y). This is especially natural since the Hermite expansion is essentially a Fourier transform up to a change of measure, and we are interested in the product of two basis functions. Let us proceed with the generating function:

$$\begin{aligned}
\mathbb{E}\left[\exp\left(\alpha\cdot x+\beta\cdot y\right)\right] &= \mathbb{E}\left[\exp\left(\alpha\cdot x+\beta\cdot\left(\rho\cdot x+\sqrt{1-\rho^2}\cdot g\right)\right)\right] \\
&= \mathbb{E}\left[\exp\left(\left(\alpha+\beta\rho\right)\cdot x\right)\cdot\exp\left(\beta\sqrt{1-\rho^2}\cdot g\right)\right] \\
&= \exp\left(\frac{1}{2}\cdot(\alpha+\beta\rho)^2\right)\cdot\exp\left(\frac{1}{2}\cdot\beta^2\cdot\left(1-\rho^2\right)\right) \\
&= \exp\left(\frac{1}{2}\cdot(\alpha^2+2\alpha\beta\rho+\beta^2)\right), \qquad (2.61)
\end{aligned}$$

where we used that generating function of Gaussian is equal to

$$\mathbb{E}\exp(t\cdot x)=\exp\left(\frac{1}{2}\cdot t^2\right).$$

Dividing (2.61) by $\exp\left(\frac{1}{2}\cdot(\alpha^2+\beta^2)\right)$ gives:

$$\mathbb{E}\left[\exp\left(\alpha\cdot x-\frac{1}{2}\cdot\alpha^2\right)\cdot\exp\left(\beta\cdot y-\frac{1}{2}\cdot\beta^2\right)\right]=\exp(\alpha\beta\rho)=\sum_{\ell=0}^{\infty}\frac{\rho^\ell}{\ell!}\cdot(\alpha\beta)^\ell. \qquad (2.62)$$

The neat point of equation (2.62) is that the terms in LHS admit a convenient form of Hermite series:

$$\exp\left(\alpha\cdot x-\frac{1}{2}\cdot\alpha^2\right)=\sum_{\ell=0}^{\infty}\frac{H_\ell(x)}{\sqrt{\ell!}}\cdot\alpha^\ell. \qquad (2.63)$$

Hence, rewriting (2.62) gives exactly the product form we need:

$$\sum_{i,j=0}^{\infty}\frac{\alpha^i\beta^j}{\sqrt{i!j!}}\cdot\mathbb{E}\left[H_i(x)H_j(y)\right]=\sum_{\ell=0}^{\infty}\frac{\rho^\ell}{\ell!}\cdot(\alpha\beta)^\ell. \qquad (2.64)$$

Since RHS and LHS of (2.64) are polynomials in (α,β) it remains to make sure that the coefficients on each of the sides "agree". This gives

$$\mathbb{E}\left[H_i(x)H_j(y)\right]=\begin{cases}\rho^i, & i=j\\ 0, & \text{otherwise,}\end{cases} \qquad (2.65)$$

which coincides with (2.59). As one might notice, the proof above also provides with another way to show "orthonormality" of Hermite polynomials by selecting $\rho=1$. Identity in (2.63) is also useful in many derivations concerning Hermite series.

CHAPTER 3

Landscape Connectivity and Dropout Stability of SGD Solutions

The optimization of multilayer neural networks typically leads to a solution with zero training error, yet the landscape can exhibit spurious local minima and the minima can be disconnected. In this chapter, we shed light on this phenomenon: we show that the combination of stochastic gradient descent (SGD) and over-parameterization makes the landscape of multilayer neural networks approximately connected and thus more favorable to optimization. More specifically, we prove that SGD solutions are connected via a piecewise linear path, and the increase in loss along this path vanishes as the number of neurons grows large. This result is a consequence of the fact that the parameters found by SGD are increasingly dropout stable as the network becomes wider. We show that, if we remove part of the neurons (and suitably rescale the remaining ones), the change in loss is independent of the total number of neurons, and it depends only on how many neurons are left. Our results exhibit a mild dependence on the input dimension: they are dimension-free for two-layer networks and require the number of neurons to scale linearly with the dimension for multilayer networks. We validate our theoretical findings with numerical experiments for different architectures and classification tasks.

3.1 Motivation and Outlook

The recent successes of deep learning have two elements in common: *(i)* a local search algorithm, e.g., stochastic gradient descent (SGD), and *(ii)* an over-parameterized neural network. Even though the training problem can have several local minima [AHW96] and is NP-hard in the worst case [BR89], the optimization of an over-parameterized network via SGD typically leads to a solution that has small training error and generalizes well. This fact has led to a focus on the theoretical understanding of neural networks' optimization landscape (see, e.g., [LSSS14, DPG$^+$14, SS16, PB17] and the discussion in Section 3.2). However, most of the existing results either make strong assumptions on the model or do not provide a satisfactory scaling with respect to the parameters of the problem.

From the empirical viewpoint, it has been observed that, if we connect two minima of SGD with a line segment, the loss is large along this path [GVS15, KMN$^+$17]. However, if the path is chosen in a more sophisticated way, one can connect the minima found by SGD via a piecewise linear path where the loss is approximately constant [GIP$^+$18, DVSH18]. These findings suggest that the minima of SGD are not isolated points in parameter space, but rather they are approximately connected. In the paper [KWL+19], mode connectivity of multilayer ReLU networks is proved by assuming generic properties of well-trained networks, i.e., dropout stability and noise stability.

In this work, we consider multilayer neural networks trained by one-pass (or online) SGD with the square loss. We show that, as the number of neurons increases, *(i)* the neural network becomes increasingly dropout stable, and *(ii)* the optimization landscape becomes increasingly connected between SGD solutions. We establish quantitative bounds on how much the loss changes after the dropout procedure and along the path connecting two SGD solutions, and we relate this change in loss to the total number of neurons, the size of the dropout pattern, and the input dimension. By doing so, we give a theoretical justification to the empirical observation that the barriers between local minima tend to disappear as the neural network becomes larger [DVSH18]. More specifically, our main contributions can be summarized as follows:

Two-layer networks. We consider the training of a two-layer neural network $\hat{y}(\boldsymbol{x}) = \frac{1}{N}\boldsymbol{a}^{\mathsf{T}}\sigma(\boldsymbol{W}\boldsymbol{x})$ with N neurons. First, we study the dropout stability of SGD solutions, namely, we bound the change in loss when $N - M$ neurons are removed from the trained network and M remaining neurons are suitably rescaled: *we show that the change in loss scales at most as $\sqrt{\log M/M}$, and therefore it does not depend on the number of neurons N of the original network or on the dimension d of the input.* Then, we characterize the landscape connectivity for the parameters obtained via SGD: *we show that pairs of SGD solutions are connected via a piecewise linear path, and the loss along this path is no larger than the loss at the extremes plus a term that scales as $\sqrt{\log N/N}$.* Let us emphasize that the two solutions of SGD are obtained by running the algorithm on different samples (from the same data distribution), for different initializations, and for the different number of iterations.

Multilayer networks. We consider the training of a general model of deep neural network with $L + 1 \geq 4$ layers, where each hidden layer contains N neurons. This model includes as a special case $\hat{\boldsymbol{y}}(\boldsymbol{x})$ which is equal to

$$\frac{1}{N}\boldsymbol{W}_{L+1}\sigma_L\left(\cdots\left(\frac{1}{N}\boldsymbol{W}_2\sigma_1\left(\boldsymbol{W}_1\boldsymbol{x}\right)\right)\cdots\right) \tag{3.1}$$

Our results are similar to those for two-layer networks: *(i) if we keep at least M neurons in each layer, the change in loss scales at most as $\sqrt{(d + \log M)/M}$; (ii) pairs of SGD solutions are connected via a piecewise linear path, along which the loss does not increase more than $\sqrt{(d + \log N)/N}$.* In contrast with the two-layer case, these bounds are not dimension-free. However, the dependence on the input dimension d is only linear, since the loss change vanishes as soon as $M, N \gg d$. We assume that, during SGD training, the parameters of the first and last layer are kept fixed, and they are regarded as random features [RR08]. We believe that this assumption, as well as the requirement of having at least 4 layers, can be removed with an improved analysis.

The proofs of dropout stability build on recent results concerning the mean-field description of the SGD dynamics [MMM19, AOY19], see also the discussion in Section 3.2. The proofs of landscape connectivity use ideas from [KWL+19].

Organization of the Chapter 3. In Section 3.2, we succinctly review related work. In Section 3.3, we present our rigorous results for two-layer networks: we first assume that the activation function σ is bounded, and then we provide an extension to unbounded activations.

In Section 3.4, we present our results for multilayer networks. In Section 3.5, we validate our findings with numerical experiments on fully-connected neural networks trained on MNIST and CIFAR-10 datasets. Finally, in Section 3.6 we discuss additional connections to the literature and give directions for future work. All the proofs are deferred to Appendix A, which also contains additional numerical results.

Notation. We use bold symbols for vectors $\boldsymbol{a}, \boldsymbol{b}$, and capitalized bold symbols for matrices $\boldsymbol{A}, \boldsymbol{B}$. We denote by $\|\boldsymbol{a}\|_2$ the norm of $\boldsymbol{a}$, by $\|\boldsymbol{A}\|_{op}$ the operator norm of $\boldsymbol{A}$, by $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ the scalar product of $\boldsymbol{a}, \boldsymbol{b}$, and by $\boldsymbol{a} \odot \boldsymbol{b}$ the Hadamard (or entrywise) product of $\boldsymbol{a}, \boldsymbol{b}$. Given an integer N and a real number $r \geq 1$, we set $[N] = \{1, \dots, N\}$ and $[r] = \{1, \dots, \lfloor r \rfloor\}$. Given a discrete set $\mathcal{A}$, we denote by $|\mathcal{A}|$ its cardinality.

3.2 Related Work

The landscape of several non-convex optimization problems has been studied in recent years, including empirical risk minimization [MBM18], low rank matrix problems [GJZ17], matrix completion [GLM16], and semi-definite programs [BVB16]. Motivated by the extraordinary success of deep learning, a growing literature is focusing on the loss surfaces of neural networks. Under strong assumptions, in [CHM+15] the loss function is related to a spin glass and it is shown that local minima are located in a well-defined band. It has been shown that local minima are globally optimal in various settings: deep linear networks [Kaw16]; fully connected and convolutional neural networks with a wide layer containing more neurons than training samples [NH17, NH18]; deep networks with more neurons than training samples and skip connections [NMH19]. Furthermore, if one of the layers is sufficiently wide, in [Ngu19b] it is shown that sublevel sets are connected. Similar results are proved for binary classification in [LSLS18a, LSLS18b]. In [FB17], a two-layer neural networks with ReLU activations is considered, and it is shown that the landscape becomes approximately connected as the number of neurons increases. However, the energy gap scales exponentially with the input dimension. In [VBB19], it is shown that there are no spurious valleys when the number of neurons is larger than the intrinsic dimension of the networks. However, for many standard architectures, the intrinsic dimension of the network is infinite.

In this chapter, we take a different view and relate the problem to a recent line of work, which shows that the behavior of neural networks trained by SGD tends to a mean field limit, as the number of neurons grows. This phenomenon has been first studied in two-layer neural networks in [MMN18, RVE18, CB18, SS18b]. In particular, in [MMN18], it is shown that the SGD dynamics is well approximated by a Wasserstein gradient flow, given that the number of neurons exceeds the data dimension. Improved and dimension-free bounds are provided in [MMM19]. Convergence to the global optimum is proved for noisy SGD in [MMN18, CB18], without any explicit rate. A convergence rate which is exponential and dimension-free is proved in [JMM20] by exploiting the displacement convexity of the limit dynamics. An argument indicating convergence in a time polynomial in the dimension is provided in [WLLM18], but for a different type of continuous flow. Fluctuations around the mean field limit are also studied in [RVE18, SS19b]. The multilayer case is tackled in [Ngu19a, SS19a, AOY19, NP23]. In [SS19a], it is considered a (less natural) model where the number of neurons grows one layer at a time. In [Ngu19a], a formalism is developed to describe the mean field limit, but the results are not rigorous. Rigorous bounds between the SGD dynamics and a limit stochastic process are established in [AOY19], where it is assumed that the first and last layer are not trained to simplify the analysis. A different approach based on the concept of neuronal embedding is put forward in [NP23]. In [NP23], it is also provided a convergence result for three-layer networks, later generalized in the companion paper [NP21].

In a nutshell, existing mean-field analyses show that the dynamics of SGD is close to a limit stochastic process. However, the consequences of this fact remain largely unexplored, since the limit process is hard to analyze. In this work, we advance the mean-field theory of neural networks, and we provide the first theoretical guarantees on two phenomena widely observed in practice: dropout stability and mode connectivity of SGD solutions.

We remark that the mean-field regime considered in this chapter is different from the "lazy training" regime that has recently received a lot of attention [AZLL19, AZLS19, COB19, DLL$^+$18, DZPS19, JGH18, LL18, ZCZG20]. In fact, in order to prove convergence of gradient descent in the lazy regime, it is crucially exploited that the parameters stay bounded in a certain region. On the contrary, in the mean field regime, the scaling of the gradient (see Eqs. (3.4) and (3.12)) ensures that the parameters move away from the initialization. The connection between the mean-field and the lazy regime is investigated in Section 4 of [MMM19] and in [CCGZ20]. We highlight that neural networks trained in the mean-field regime achieve results comparable to the state of the art for standard datasets, as demonstrated in the numerical results of Section 3.5.

3.3 Dropout Stability and Connectivity for Two-Layer Neural Networks

3.3.1 Setup

We consider a two-layer neural network with N neurons:

$$\hat{y}_N(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} a_i \sigma(\boldsymbol{x}, \boldsymbol{w}_i), \tag{3.2}$$

where $\boldsymbol{x} \in \mathbb{R}^d$ is a feature vector, $\hat{y}_N(\boldsymbol{x}, \boldsymbol{\theta}) \in \mathbb{R}$ is the output of the network, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N)$, with $\boldsymbol{\theta}_i = (a_i, \boldsymbol{w}_i) \in \mathbb{R}^{D+1}$, are the parameters of the network and $\sigma : \mathbb{R}^d \times \mathbb{R}^D \to \mathbb{R}$ is an activation function. We remark that (3.2) is precisely the mean-field version of a two-layer network model discussed in Chapter 2 and defined as per equation (2.25).

A typical example is $\sigma(\boldsymbol{x}, \boldsymbol{w}) = \sigma(\langle \boldsymbol{x}, \boldsymbol{w} \rangle)$, for a scalar function $\sigma : \mathbb{R} \to \mathbb{R}$. In order to incorporate a bias term in the hidden layer, one can simply add the feature 1 to $\boldsymbol{x}$ and adjust the shape of the parameters $\boldsymbol{w}_i$ accordingly. We are interested in minimizing the expected square loss (also known as population risk):

$$L_N(\boldsymbol{\theta}) = \mathbb{E}\Big\{\big(y - \hat{y}_N(\boldsymbol{x}, \boldsymbol{\theta})\big)^2\Big\}, \tag{3.3}$$

where the expectation is taken over $(\boldsymbol{x}, y) \sim \mathbb{P}$. To do so, we are given data $(\boldsymbol{x}_k, y_k)_{k \geq 0} \overset{\text{i.i.d.}}{\sim} \mathbb{P}$, and we learn the parameters of the network via stochastic gradient descent (SGD) with step size s_k:

$$\begin{aligned} \boldsymbol{\theta}_i^{k+1} &= \boldsymbol{\theta}_i^k - s_k N \cdot \mathrm{Grad}_i(\boldsymbol{\theta}^k), \\ \mathrm{Grad}_i(\boldsymbol{\theta}^k) &= \nabla_{\boldsymbol{\theta}_i} \big(y_k - \hat{y}_N(\boldsymbol{x}_k, \boldsymbol{\theta}^k)\big)^2, \end{aligned} \tag{3.4}$$

where $\boldsymbol{\theta}^k$ denotes the parameters after k steps of SGD, and the parameters are initialized independently according to the distribution ρ_0. We consider a one-pass (or online) model, where each data point is used only once.

Given a neural network with parameters $\boldsymbol{\theta}$ and a subset $\mathcal{A}$ of $[N]$, the dropout network with parameters $\boldsymbol{\theta}_{\rm S}$ is obtained by setting to 0 the outputs of the neurons indexed by $[N] \setminus \mathcal{A}$ and by suitably rescaling the remaining outputs. Denote by $\hat{y}_{|\mathcal{A}|}(\boldsymbol{x}, \boldsymbol{\theta}_{\rm S})$ and $L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S})$ the output of the dropout network and its expected square loss, respectively. In formulas,

$$\begin{aligned} \hat{y}_{|\mathcal{A}|}(\boldsymbol{x}, \boldsymbol{\theta}_{\rm S}) &= \frac{1}{|\mathcal{A}|}\sum_{i\in\mathcal{A}} a_i \sigma(\boldsymbol{x}, \boldsymbol{w}_i), \\ L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}) &= \mathbb{E}\left\{\left(y - \hat{y}_{|\mathcal{A}|}(\boldsymbol{x}, \boldsymbol{\theta}_{\rm S})\right)^2\right\}. \end{aligned} \tag{3.5}$$

Let us compare the original network (3.2) with the dropout network (3.5): $\boldsymbol{w}_i$ does not change, a_i is rescaled by $|\mathcal{A}|/|N|$ and in (3.5) we sum over $|\mathcal{A}|$ neurons (while in (3.2) the sum is over N neurons). This is equivalent to setting $|N| - |\mathcal{A}|$ neurons to zero and rescaling the others by a factor, as in [KWL$^+$19].

We now define the notions of dropout stability and connectivity for network parameters.

Definition 3.3.1 (Dropout stability). *Given $\mathcal{A} \subseteq [N]$, we say that $\boldsymbol{\theta}$ is $\varepsilon_{\rm D}$-dropout stable if*

$$|L_N(\boldsymbol{\theta}) - L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S})| \le \varepsilon_{\rm D}. \tag{3.6}$$

Definition 3.3.2 (Connectivity). *We say that two parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are $\varepsilon_{\rm C}$-connected if there exists a continuous path in parameter space $\pi : [0,1] \to \mathbb{R}^{D\times N}$, such that $\pi(0) = \boldsymbol{\theta}$ and $\pi(1) = \boldsymbol{\theta}'$ with*

$$L_N(\pi(t)) \le \max(L_N(\boldsymbol{\theta}), L_N(\boldsymbol{\theta}')) + \varepsilon_{\rm C}. \tag{3.7}$$

3.3.2 Results for Bounded Activations

We make the following assumptions on the learning rate s_k, the data distribution $(\boldsymbol{x}, y) \sim \mathbb{P}$, the activation function σ, and the initialization ρ_0:

(A1) $s_k = \alpha\xi(k\alpha)$, where $\xi : \mathbb{R}_{\ge 0} \to \mathbb{R}_{>0}$ is bounded by K_1 and K_1-Lipschitz.
(A2) The response variables y are bounded by K_2 and the gradient $\nabla_{\boldsymbol{w}}\sigma(\boldsymbol{x}, \boldsymbol{w})$ is K_2 sub-gaussian when $\boldsymbol{x} \sim \mathbb{P}$.
(A3) The activation function σ is bounded by K_3 and differentiable, with gradient bounded by K_3 and K_3-Lipschitz.
(A4) The initialization ρ_0 is supported on $|a_i^0| \le K_4$.

We are now ready to present our results, which are proved in Appendix A.1.

Theorem 1 (Two-layer). *Assume that conditions **(A1)**-**(A4)** hold, and fix $T \ge 1$. Let $\boldsymbol{\theta}^k$ be obtained by running k steps of the SGD algorithm* (3.4) *with data $\{(\boldsymbol{x}_j, y_j)\}_{j=0}^k \overset{\text{i.i.d.}}{\sim} \mathbb{P}$ and initialization ρ_0. Then, the following results hold:*

(A) *Pick $\mathcal{A} \subseteq [N]$ independent of $\boldsymbol{\theta}^k$. Then, with probability at least $1 - e^{-z^2}$, for all $k \in [T/\alpha]$, $\boldsymbol{\theta}^k$ is $\varepsilon_{\rm D}$-dropout stable with $\varepsilon_{\rm D}$ equal to*

$$Ke^{KT^3}\left(\frac{\sqrt{\log|\mathcal{A}|} + z}{\sqrt{|\mathcal{A}|}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right), \tag{3.8}$$

where the constant K depends only on the constants K_i of the assumptions.

(B) *Fix $T' \geq 1$ and let $(\boldsymbol{\theta}')^{k'}$ be obtained by running k' steps of SGD with data $\{(\boldsymbol{x}'_j, y'_j)\}_{j=0}^{k'} \overset{\text{i.i.d.}}{\sim} \mathbb{P}$ and initialization ρ'_0 that satisfies **(A4)**. Then, with probability at least $1 - e^{-z^2}$, for all $k \in [T/\alpha]$ and $k' \in [T'/\alpha]$, $\boldsymbol{\theta}^k$ and $(\boldsymbol{\theta}')^{k'}$ are ε_{C}-connected with ε_{C} equal to*

$$Ke^{KT_{\max}^3}\left(\frac{\sqrt{\log N} + z}{\sqrt{N}} + \sqrt{\alpha}\Big(\sqrt{D + \log N} + z\Big)\right), \tag{3.9}$$

where $T_{\max} = \max(T, T')$. Furthermore, the path connecting $\boldsymbol{\theta}^k$ with $(\boldsymbol{\theta}')^{k'}$ consists of 7 line segments.

The result **(A)** characterizes the change in loss when only $|\mathcal{A}|$ neurons remain in the network. In particular, the change in loss scales as $\sqrt{\log|\mathcal{A}|/|\mathcal{A}|} + \sqrt{\alpha(D + \log N)}$, where N is the total number of neurons, D is the dimension of the neurons and α is the step size of SGD. This quantity vanishes as long as $|\mathcal{A}| \gg 1$ and $\alpha \ll 1/(D + \log N)$. Note that the number of training samples k is such that $k\alpha$ is a constant. Thus, the condition $\alpha \ll 1/(D + \log N)$ implies that k needs to scale only logarithmically with N. Furthermore, the condition $|\mathcal{A}| \gg 1$ implies that $|\mathcal{A}|$ does not need to scale with N, D. The proof builds on the machinery developed in [MMM19] to provide a mean-field approximation to the dynamics of SGD. In [MMM19], it is shown that, as $N \to \infty$ and $\alpha \to 0$, the parameters $\boldsymbol{\theta}^k$ obtained by running k steps of SGD with step size α are close to N i.i.d. particles that evolve according to a nonlinear dynamics at time $k\alpha$. Here, the idea is to show that *(i)* the parameters $\boldsymbol{\theta}_{\mathrm{S}}^k$ are also close to $|\mathcal{A}|$ such i.i.d. particles, and *(ii)* the quantities $L_N(\boldsymbol{\theta}^k)$ and $L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}^k)$ concentrate to the same limit value, which represents the limit loss of the nonlinear dynamics.

The result **(B)** shows that we can connect two different solutions of SGD via a simple path. Note that the two solutions can be obtained by running SGD for the different number of iterations ($k' \neq k$), for different training datasets ($(\boldsymbol{x}_j, y_j) \neq (\boldsymbol{x}'_j, y'_j)$) and for different initializations of SGD ($\rho_0 \neq \rho'_0$). The proof uses ideas from [KWL+19]. In that work, the authors consider a multilayer neural network with ReLU activations and show how to find a piecewise linear path between two solutions that are dropout stable with $|\mathcal{A}| = N/2$. In fact, ε_{C} has a similar scaling to ε_{D} after setting $|\mathcal{A}| = N/2$. We are also able to show (and, consequently, exploit) a more general notion of dropout stability for the trained network. In fact, [KWL+19] requires the existence of a single dropout pattern, while here we give a bound for any fixed dropout pattern (as long as it does not depend on SGD).

The bounds in Theorem 1 exhibit an exponential dependence on T. We remark that, in the mean-field regime, the number of samples k is large, the step size α is small, and $T = k\alpha$ is a constant. In fact, T is the evolution time of the limit stochastic process (which does not depend on N, α). Empirically, the value of T needed to achieve good accuracy is quite small: $T = 1$ gives $< 16\%$ error on CIFAR-10, see Section 3.5. The exponential dependence on T is common to all existing mean-field analyses, and improving it is an open question. The assumptions on the learning rate, the data distribution and the initialization are mild and only require some regularity. The assumptions on the activation function are fulfilled in several practical settings: $\sigma(\boldsymbol{x}, \boldsymbol{w}) = \sigma(\langle \boldsymbol{x}, \boldsymbol{w} \rangle)$, where $\sigma : \mathbb{R} \to \mathbb{R}$ is, e.g., the sigmoid or the hyperbolic tangent.

3.3.3 Extension to Unbounded Activations

Note that Theorem 1 requires that the activation function is bounded. We can relax this assumption, at the cost of a less tight dependence on the time T of the evolution. In particular, assume further that *(i)* the feature vectors $\boldsymbol{x}$ and the initialization ρ_0 are bounded, and that *(ii)* the loss at each step of SGD is uniformly bounded, i.e., $\max_j |y_j - \hat{y}_N(\boldsymbol{x}_j, \boldsymbol{\theta}^j)| \le K_5$. This last requirement is reasonable, since the objective of SGD is to minimize such a loss. Then, the results of Theorem 1 hold also for unbounded σ, where the term Ke^{KT^3} is replaced by a generic $K(T)$, which depends on T and on the constants K_i of the assumptions. The simulation results of Section 3.5 show that such a dependence on T is mild in practical settings.

The formal statement and the proof of this result is contained in Appendix A.2. The idea is to show that, if the parameters of the neural network are initialized with a bounded distribution, then they stay bounded for any finite time T of the SGD evolution. Thus, the SGD evolution does not change if we substitute the unbounded activation function with a bounded one, and we can apply the results for bounded σ.

3.4 Dropout Stability and Connectivity for Multilayer Neural Networks

3.4.1 Setup

We consider a neural network with $L+1 \ge 4$ layers, where each hidden layer contains N neurons. Given the input feature vector $\boldsymbol{x} \in \mathbb{R}^{d_0}$, the first layer activations $\boldsymbol{z}^{(1)}_{i_1}$ for $i_1 \in [N]$ have the form

$$\sigma^{(0)}\left(\boldsymbol{x}, \boldsymbol{\theta}^{(0)}_{i_1}\right), \quad \boldsymbol{\theta}^{(0)}_{i_1} \in \mathbb{R}^{D_0}$$

the intermediate layer $\ell \in [L-1]$ activations $\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}(\boldsymbol{x}, \boldsymbol{\theta})$ for $i_{\ell+1} \in [N]$ are defined as follows

$$\begin{aligned} &\frac{1}{N}\sum_{i_\ell=1}^{N} \boldsymbol{a}^{(\ell)}_{i_\ell, i_{\ell+1}} \odot \sigma^{(\ell)}\left(\boldsymbol{z}^{(\ell)}_{i_\ell}(\boldsymbol{x}, \boldsymbol{\theta}), \boldsymbol{w}^{(\ell)}_{i_\ell, i_{\ell+1}}\right), \\ &\boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}} = (\boldsymbol{a}^{(\ell)}_{i_\ell, i_{\ell+1}}, \boldsymbol{w}^{(\ell)}_{i_\ell, i_{\ell+1}}) \in \mathbb{R}^{D_\ell + d_{\ell+1}}, \end{aligned}$$

and the output of the network is given by

$$\begin{aligned} \hat{\boldsymbol{y}}_N(\boldsymbol{x}, \boldsymbol{\theta}) &= \frac{1}{N}\sum_{i_L=1}^{N} \boldsymbol{a}^{(L)}_{i_L} \odot \sigma^{(L)}\left(\boldsymbol{z}^{(L)}_{i_L}(\boldsymbol{x}, \boldsymbol{\theta}), \boldsymbol{w}^{(L)}_{i_L}\right), \\ \boldsymbol{\theta}^{(L)}_{i_L} &= (\boldsymbol{a}^{(L)}_{i_L}, \boldsymbol{w}^{(L)}_{i_L}) \in \mathbb{R}^{D_L + d_{L+1}}, \quad i_L \in [N]. \end{aligned} \tag{3.10}$$

Here, $\sigma^{(\ell)} : \mathbb{R}^{d_\ell} \times \mathbb{R}^{D_\ell} \to \mathbb{R}^{d_{\ell+1}}$ ($\ell \in \{0, \dots, L\}$) are the activation functions, and $\boldsymbol{\theta}$ contains the parameters of the network, which are $\boldsymbol{\theta}^{(0)}_{i_1}$, $\boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}}$ and $\boldsymbol{\theta}^{(L)}_{i_L}$.

Note that (3.10) includes the model (3.1) as a special case. To see this, consider the following setting: pick $D_0 = d_0$ and stack the parameters $\boldsymbol{\theta}^{(0)}_{i_1} \in \mathbb{R}^{d_0}$ into the rows of the matrix $\boldsymbol{W}_1 \in \mathbb{R}^{N \times d_0}$; for $i \in [L-1]$, pick $D_\ell = 1$ and stack the scalar parameters $\boldsymbol{a}^{(\ell)}_{i_\ell, i_{\ell+1}} \in \mathbb{R}$ into the matrix $\boldsymbol{W}_{\ell+1} \in \mathbb{R}^{N \times N}$; pick $D_L = d_{L+1}$ and stack the parameters $\boldsymbol{a}^{(L)}_{i_L} \in \mathbb{R}^{d_{L+1}}$ into the columns of the matrix $\boldsymbol{W}_{L+1} \in \mathbb{R}^{d_{L+1} \times N}$; finally, assume that the activation function $\sigma^{(\ell)}$ does not depend on $\boldsymbol{w}^{(\ell)}_{i_\ell, i_{\ell+1}}$ for $\ell \in [L-1]$ and that $\sigma^{(L)}$ does not depend on $\boldsymbol{w}^{(L)}_{i_L}$. Then, in this setting, (3.10) can be reduced to (3.1).

We are interested in minimizing the expected square loss:

$$L_N(\boldsymbol{\theta}) = \mathbb{E}\left\{ \left\| \boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}\right) \right\|_2^2 \right\}, \tag{3.11}$$

where the expectation is taken over $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathbb{P}$. To do so, we are given data $(\boldsymbol{x}_k, \boldsymbol{y}_k)_{k\geq 0} \overset{\text{i.i.d.}}{\sim} \mathbb{P}$, we run SGD with step size s_k for the intermediate layers $\ell \in [L-1]$, and we fix first and last layer:

$$\begin{aligned}
&\boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}}(k+1) = \boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}}(k) - s_k N^2 \mathrm{Grad}^{(\ell)}_{i_\ell, i_{\ell+1}}\Big(\boldsymbol{\theta}(k)\Big), \\
&\mathrm{Grad}^{(\ell)}_{i_\ell, i_{\ell+1}}\Big(\boldsymbol{\theta}(k)\Big) = \nabla_{\boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}}} \left\| \boldsymbol{y}_k - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_k, \boldsymbol{\theta}(k)\right) \right\|_2^2, \\
&\boldsymbol{\theta}^{(0)}_{i_1}(k+1) = \boldsymbol{\theta}^{(0)}_{i_1}(k), \quad \boldsymbol{\theta}^{(L)}_{i_L}(k+1) = \boldsymbol{\theta}^{(L)}_{i_L}(k),
\end{aligned} \tag{3.12}$$

where $\boldsymbol{\theta}(k)$ contains the parameters of the network after k steps of SGD. As in the two-layer setting, we consider a one-pass model and the parameters are initialized independently, i.e., $\{\boldsymbol{\theta}^{(0)}_{i_1}(0)\}_{i_1\in[N]} \overset{\text{i.i.d.}}{\sim} \rho_0^{(0)}$, $\{\boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}}(0)\}_{i_\ell, i_{\ell+1}\in[N]} \overset{\text{i.i.d.}}{\sim} \rho_0^{(\ell)}$, for $\ell \in [L-1]$, and $\{\boldsymbol{\theta}^{(L)}_{i_L}(0)\}_{i_L\in[N]} \overset{\text{i.i.d.}}{\sim} \rho_0^{(L)}$.

The gradients of $\widehat{\boldsymbol{y}}_N$ with respect to the parameters of the network can be computed via backpropagation [GBC16]. By doing so (see [AOY19, Section 3.3]), we obtain that $\boldsymbol{\theta}^{(\ell)}_{i_\ell, i_{\ell+1}}$ evolves at a time scale of $1/N^2$. Thus, we multiply the step size s_k in (3.12) with the factor N^2 in order to avoid falling into the "lazy training" regime. In lazy training, the parameters hardly vary but the method still converges to zero training loss, and this regime has received a lot of attention recently [JGH18, LL18, ZCZG20, DLL$^+$18, DZPS19, AZLS19, AZLL19, COB19]. Let us emphasize that the SGD scalings in (3.4) and (3.12) imply that the parameters move as long as the product of the number of iterations with the step size is non-vanishing.

Note also that the parameters of layers $\ell = 0$ and $\ell = L$, i.e., $\{\boldsymbol{\theta}^{(0)}_{i_1}\}_{i_1\in[N]}$ and $\{\boldsymbol{\theta}^{(L)}_{i_L}\}_{i_L\in[N]}$, stay fixed to their initial values. This is done for technical reasons. In fact, by computing the backpropagation equations, one obtains that $\boldsymbol{\theta}^{(0)}_{i_1}$ and $\boldsymbol{\theta}^{(L)}_{i_L}$ evolve at a time scale of $1/N$, which makes it challenging to analyze their trajectories. We regard the parameters $\boldsymbol{\theta}^{(0)}_{i_1}$ and $\boldsymbol{\theta}^{(L)}_{i_L}$ as random features [RR08] close to the input and the output.

Given a neural network with parameters $\boldsymbol{\theta}$ and subsets $\mathcal{A}_1, \dots, \mathcal{A}_L$ of $[N]$, the dropout network with parameters $\boldsymbol{\theta}_{\mathrm{S}}$ is obtained by setting to 0 the outputs of the neurons indexed by $[N] \setminus \mathcal{A}_i$ at layer i and by suitably rescaling the remaining outputs. With an abuse of notation, denote by $\widehat{\boldsymbol{y}}_{|\mathcal{A}|}\left(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}\right)$ and $L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}})$ the output of the dropout network and its expected square loss, respectively. In formulas, the dropout version of activations $\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}\left(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}\right)$ of layer $\ell \in [L-1]$ for $i_{\ell+1} \in \mathcal{A}_{\ell+1}$ are given by

$$\frac{1}{|\mathcal{A}_\ell|} \sum_{i_\ell \in \mathcal{A}_\ell} \boldsymbol{a}^{(\ell)}_{i_\ell, i_{\ell+1}} \odot \sigma^{(\ell)}\left(\boldsymbol{z}^{(\ell)}_{i_\ell}\left(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}\right), \boldsymbol{w}^{(\ell)}_{i_\ell, i_{\ell+1}} \right),$$

the output of dropout network $\widehat{\boldsymbol{y}}_{|\mathcal{A}|}\left(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}\right)$ takes the form

$$\frac{1}{|\mathcal{A}_L|} \sum_{i_L \in \mathcal{A}_L} \boldsymbol{a}^{(L)}_{i_L} \odot \sigma^{(L)}\left(\boldsymbol{z}^{(L)}_{i_L}\left(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}\right), \boldsymbol{w}^{(L)}_{i_L} \right),$$

and, consequently, the expected square loss is defined by

$$L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}) = \mathbb{E}\left\{ \left\| \boldsymbol{y} - \widehat{\boldsymbol{y}}_{|\mathcal{A}|}\left(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}\right) \right\|_2^2 \right\},$$

where $\boldsymbol{z}_{i_1}^{(1)}(\boldsymbol{x}, \boldsymbol{\theta}_{\mathrm{S}}) = \boldsymbol{z}_{i_1}^{(1)}(\boldsymbol{x}, \boldsymbol{\theta})$ for $i_1 \in \mathcal{A}_1$. The definitions of dropout stability and connectivity are analogous to those for two-layer networks: *(i)* $\boldsymbol{\theta}$ is ε_{D}-dropout stable if (3.6) holds; and *(ii)* $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are ε_{C}-connected if they are connected by a continuous path in parameter space such that (3.7) holds.

3.4.2 Results

We make the following assumptions on the learning rate s_k, the data distribution $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathbb{P}$, the activation functions $\sigma^{(\ell)}$, and the initializations $\rho_0^{(\ell)}$:

(B1) $s_k = \alpha\xi(k\alpha)$, where $\xi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{>0}$ is bounded by K_1 and K_1-Lipschitz.
(B2) The response variables $\boldsymbol{y}$ are bounded by K_2.
(B3) For $\ell \in \{0, \ldots, L\}$, the activation function $\sigma^{(\ell)}$ is bounded by K_3, with Fréchet derivative bounded by K_3 and K_3-Lipschitz.
(B4) The initializations $\{\rho_0^{(\ell)}\}_{\ell=0}^L$ have finite first moment and they are supported on $\|\boldsymbol{a}_{i_\ell, i_{\ell+1}}^{(\ell)}(0)\|_2 \leq K_4$ for $\ell \in [L-1]$, and $\|\boldsymbol{a}_{i_L}^{(L)}(0)\|_2 \leq K_4$.

We are now ready to present our results, which are proved in Appendix A.3.

Theorem 2 (Multilayer)**.** *Assume that conditions **(B1)**-**(B4)** hold, let $\boldsymbol{\theta}(k)$ be obtained by running k steps of the SGD algorithm* (3.12) *with data $\{(\boldsymbol{x}_j, \boldsymbol{y}_j)\}_{j=0}^k \overset{\text{i.i.d.}}{\sim} \mathbb{P}$ and initializations $\{\rho_0^{(\ell)}\}_{\ell=0}^L$, and define $T = k\alpha > 0$. Then, the following results hold:*

***(A)** Pick $\mathcal{A}_1, \ldots, \mathcal{A}_L \subseteq [N]$ independent of $\boldsymbol{\theta}(k)$. Then, with probability at least $1 - e^{-z^2}$, $\boldsymbol{\theta}(k)$ is ε_{D}-dropout stable with ε_{D} equal to*

$$K(T, L)\left(\frac{\sqrt{d} + z}{\sqrt{A_{\min}}} + \sqrt{\frac{\log N}{N}} + \sqrt{\alpha}\Big(\sqrt{d + \log N} + z\Big)\right) \tag{3.13}$$

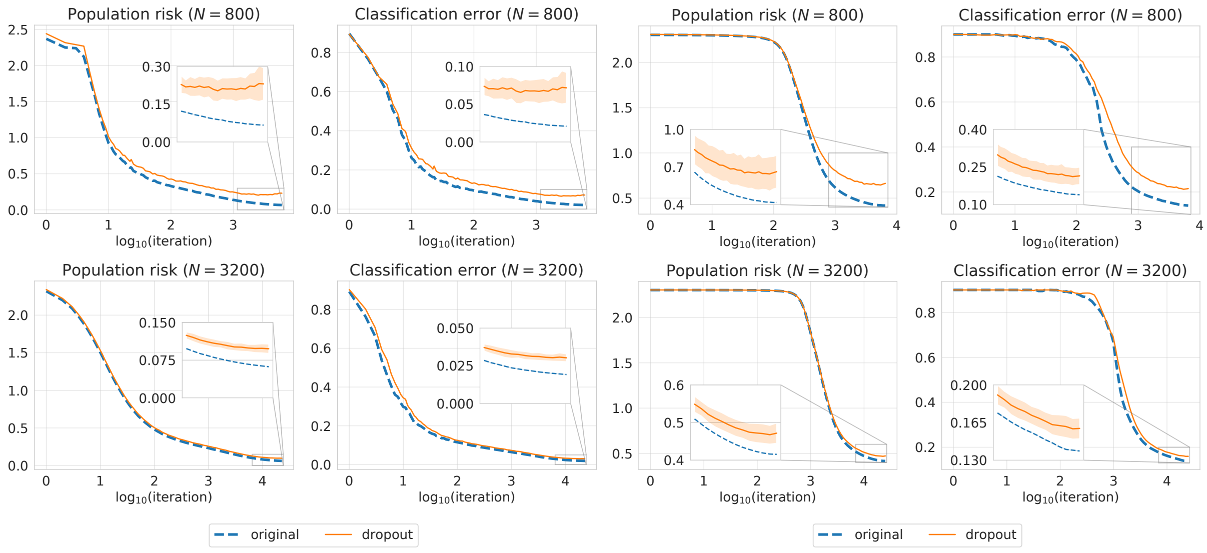
where $A_{\min} = \min_{i \in [L]} |\mathcal{A}_i|$, $d = \max_{\ell \in \{0, \ldots, L+1\}} d_\ell$ and the constant $K(T, L)$ depends on T, L and on the constants K_i of the assumptions.

***(B)** Let $\boldsymbol{\theta}'(k')$ be obtained by running k' steps of the SGD algorithm* (3.12) *with data $\{(\boldsymbol{x}_j', \boldsymbol{y}_j')\}_{j=0}^{k'} \overset{\text{i.i.d.}}{\sim} \mathbb{P}$ and initializations $\{(\rho_0^{(\ell)})'\}_{\ell=0}^L$ that satisfy **(B4)**, and define $T' = k'\alpha > 0$. Then, with probability at least $1 - e^{-z^2}$, $\boldsymbol{\theta}(k)$ and $\boldsymbol{\theta}'(k')$ are ε_{C}-connected with ε_{C} equal*

$$K(T_{\max}, L)\left(\frac{\sqrt{d + \log N} + z}{\sqrt{N}} + \sqrt{\alpha}\Big(\sqrt{d + \log N} + z\Big)\right) \tag{3.14}$$

where $T_{\max} = \max(T, T')$.

The results are similar in spirit to those of Theorem 1, but the analysis is more involved. We remark that, differently from the two-layer case, the ideal particles are not independent, see Remark 5.6 of [AOY19]. We exploit a bound on the norm of the weights during training (see



(a) MNIST, two-layer

(b) CIFAR-10, three-layer

Figure 3.1: Comparison of population risk and classification error between the trained network (blue dashed curve) and the dropout network (orange curve). In the full scale plot, we show the average values, and in the zoomed version we also provide the error bar.

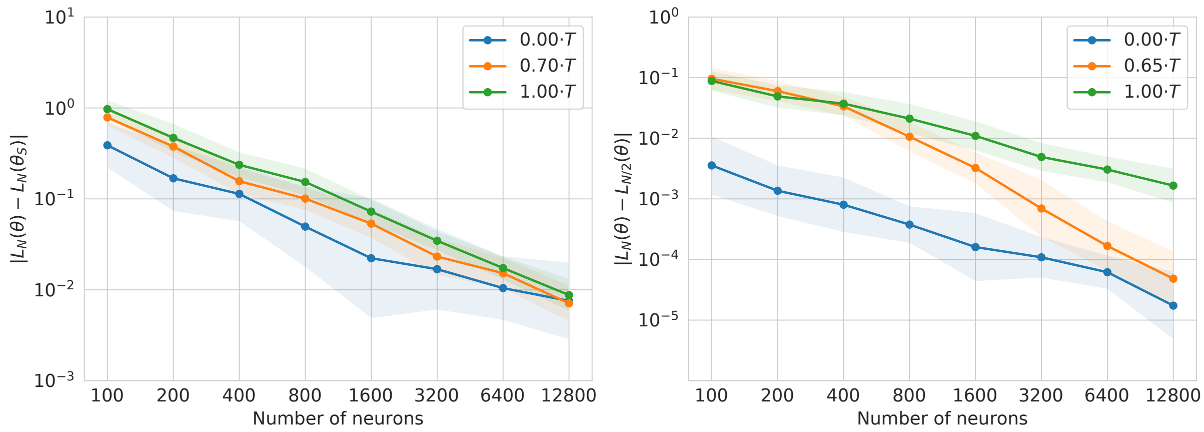
Lemma A.3.1 in Appendix A.3.1) and a bound on the *maximum* distance between SGD weights and weights of ideal particles. Our analysis improves upon [AOY19], where the bound is on the *average* distance between SGD and ideal-particle weights (compare (A.56) in Appendix A.3.1 and (10.1) in [AOY19]). This improvement is essential to show dropout stability. In fact, dropout stability requires dropping all weights associated to a subnetwork (and not just a given fraction of weights). The stronger guarantee on the distance to ideal particles leads to an extra $\log N$ in our bounds (compare Theorem 2 in this chapter and (5.1) in [AOY19]). As concerns the proof of connectivity, we generalize the approach of [KWL$^+$19], in order to analyze the model (3.10).

The bounds in Theorem 2 are not dimension-free (as in the two-layer case), but the dependence on the dimension d is only linear. In fact, the loss change in (3.13) vanishes as long as $A_{\min} \gg d$, and $\alpha \ll 1/(d + \log N)$. The condition $A_{\min} \gg d$ implies that $A_{\min}$ needs to scale at least linearly with d, but does not scale with N. Furthermore, as in the two-layer case, the condition $\alpha \ll 1/(d + \log N)$ implies that the number of samples k needs to scale only logarithmically with N.

Compared to the two-layer case where there is no assumption on the initialization for $\boldsymbol{w}_i$, here we require a mild condition (finite first moment for $\rho_0^{(\ell)}$) in order to simplify the proof.

3.5 Numerical Results

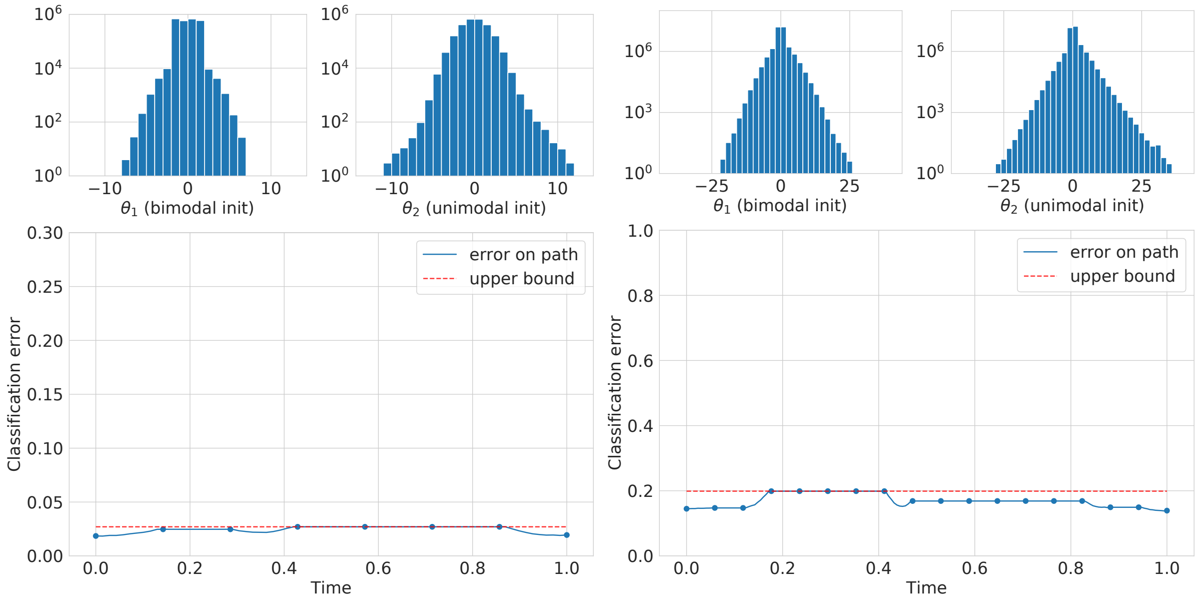
We consider two supervised learning tasks: *(a)* MNIST classification with the two-layer neural network (3.2); and *(b)* CIFAR-10 classification with the three-layer neural network (3.1). For MNIST, the input dimension is $d = 28 \times 28 = 784$ and we normalize pixel values to have zero mean and unit variance. For CIFAR-10, the input is given by VGG-16 features of dimension $d = 4 \times 4 \times 512 = 8192$. These features are computed by the convolutional layers of the VGG-16 network [SZ15] pre-trained on the ImageNet dataset [RDS$^+$15]. More specifically, we rescale the images to size 128×128, we rescale pixel values into the range $[-1, 1]$, and we



(a) MNIST, two-layer

(b) CIFAR-10, three-layer

Figure 3.2: Change in loss after removing half of the neurons from each layer, as a function of the number of neurons N of the full network.



(a) MNIST, two-layer

(b) CIFAR-10, three-layer

Figure 3.3: Classification error along a piecewise linear path that connects two SGD solutions $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, with $N = 3200$. As predicted by the theory, the error along the path (blue curve) is no larger than the error of the two SGD solutions plus the change in loss due to the dropout of half of the neurons (red dashed curve).

feed them to the pre-trained VGG-16 network to extract the features. Qualitatively similar results (with larger classification error) are obtained by using fully connected networks directly on CIFAR-10 images.

For both tasks, the neural networks have ReLU activation functions, SGD aims at minimizing the cross-entropy loss, and the gradients are averaged over mini-batches of size 100. In contrast with the setting of Section 3.4, all the layers of the neural network are trained. The scaling of the gradient updates follows (3.4) and (3.12): for the first and last layer, the gradient of the loss function is multiplied by a factor of N; for the middle layers, the gradient of the loss function is multiplied by a factor of N^2. This scaling ensures that the term in front of the learning rate s_k does not depend on N, i.e., it is $\Theta(1)$ as N goes large. The learning rate

$s_k = \alpha\xi(k\alpha)$ does not depend on the time of the evolution, i.e., $\xi(t) = 1$. Furthermore, we set $\alpha = \alpha_0/N$, where α_0 is a constant independent of N. We also set the number of training epochs to $k_0 \cdot N$, where k_0 is a constant independent of N. In this way, the product between the learning rate and the number of training epochs is the constant $T = k_0 \cdot \alpha_0$, which does not depend on N. The initializations of the parameters of the neural network are i.i.d. and do not depend on N, as in the setting described for the theoretical results. The population risk and the classification error are obtained by averaging over the test dataset. To measure statistics in the plots, i.e., average value and error bar at 1 standard deviation, we perform 20 independent trials of each experiment.

Figure 3.1 compares the performance of the trained network (blue dashed curve) and of the dropout network (orange curve), which is obtained by removing the second half of the neurons from each layer (and by suitably rescaling the remaining neurons). On the left, we report the results for MNIST, and on the right for CIFAR-10. For each classification task, we plot the population risk and the classification error for $N = 800$ and $N = 3200$. The networks are trained until the training loss has reached a plateau (0.062 for MNIST and 0.415 for CIFAR-10 when $N = 3200$). As expected, the performance of the dropout network improves with N, and it is very close to that of the trained network. For $N = 3200$, the classification error of the trained network is $< 2\%$ for MNIST and $< 14\%$ on CIFAR-10, and the classification error of the dropout network is $\approx 3\%$ on MNIST and $< 16\%$ on CIFAR-10.

Figure 3.2 plots the change in loss when only half of the neurons remain in the network, as a function of the total number of neurons N. For each classification task, we plot the change in loss at the beginning of training $(0 \cdot T)$, at an intermediate point where the population risk is still not too small $(\{0.65, 0.7\} \cdot T)$, and at the end of training $(1 \cdot T)$, where T stands for the product of the learning rate and the total number of training epochs. The dependence between the change in loss and N is essentially linear in log-log scale, as demonstrated by our theoretical results. Furthermore, the dependence on the time of the dynamics is quite mild.

Figure 3.3 shows that the optimization landscape is approximately connected when $N = 3200$. We plot the classification error along a piecewise linear path that connects two SGD solutions $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ initialized with different distributions: the initial distribution of $\boldsymbol{\theta}_1$ is bimodal, while the initial distribution of $\boldsymbol{\theta}_2$ is unimodal. We also show the histograms of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, in order to highlight that one SGD solution cannot be obtained as a permutation of the other. As expected, the classification error along the path is roughly constant, since the network is dropout stable. More specifically, the error along the path (blue curve) is upper bounded by the error at the extremes plus the change in loss after dropping out half of the neurons of the network (red dashed curve).

Figure 3.4 plots the degradation in classification error due to the removal of half of the neurons from each layer. We consider neural networks at the end of training $(1 \cdot T)$ and we report the performance degradation as a function of the number of neurons N of the full network. We compare different architectures (two-layer, three-layer and four-layer neural networks) and classification tasks (MNIST and CIFAR-10). In all the cases considered, the performance degradation rapidly decreases, as the width of the network grows. When $N = 12800$, the classification error increases only *(i)* by 0.35% for a two-layer network trained on MNIST, *(ii)* by 0.4% for a three-layer network trained on MNIST, *(iii)* by 1% for a three-layer network trained on CIFAR-10, and *(iv)* by 3.6% for a four-layer network trained on CIFAR-10.

Additional experiments are presented in Appendix A.4 for the following learning tasks: classification of isotropic Gaussians with the two-layer neural network (3.2); MNIST classification

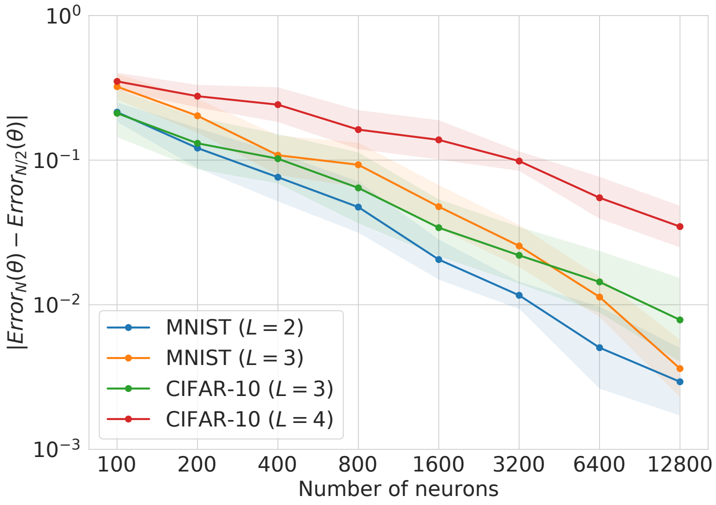


Figure 3.4: Change in classification error after removing half of the neurons from each layer, as a function of the number of neurons N of the full network, at the end of training.

with the three-layer neural network (3.1); CIFAR-10 classification with the four-layer neural network (3.1).

3.6 Discussion and Future Directions

The optimization landscape of neural networks can exhibit spurious local minima [YSJ18, SS18a], and its minima can be disconnected [FB17, VBB19, KWL+19]. In this work, we show that these problematic scenarios are ruled out with SGD training and over-parametrization. In particular, we prove that the optimization landscape of SGD solutions is increasingly connected as the number of neurons grows. The explanation to this phenomenon has been hypothesized by some recent work: the SGD solutions have degrees of freedom to spare [DVSH18] or, equivalently, they are dropout stable [KWL+19]. We give theoretical grounding to this conjecture by proving that SGD solutions are dropout stable, i.e., that the loss does not change much when we remove even a large amount of neurons. In order to have meaningful bounds, the number of neurons does not need to be of the same order of the number of samples (cf. [NH17, NH18, NMH19, Ngu19b]). Furthermore, our bounds are dimension-free for two-layer networks and they scale linearly with the dimension for multilayer networks (cf. [FB17]). Our analysis builds on a recent line of work showing that the dynamics of SGD tends to a mean field limit as the number of neurons increases [MMN18, MMM19, AOY19]. We believe that with these tools one could prove similar results also for noisy SGD and projected SGD.

The notion of dropout stability is closely related to the fact that neural networks have many redundant connections, and therefore they can be pruned with little performance loss, see, e.g., [GYC16, MAV17, FC19, LSZ+19]. However, it is difficult even to compare the relative merits of the different pruning techniques [GEH19], let alone to understand the fundamental reasons leading to sparsity in neural networks. Thus, it would be interesting to investigate whether mean field approaches provide a more principled way of pruning deep neural networks.

CHAPTER 4

Mean-field Analysis of Piecewise Linear Solutions for Wide ReLU Networks

Understanding the properties of neural networks trained via stochastic gradient descent (SGD) is at the heart of the theory of deep learning. In this chapter, we take a mean-field view, and consider a two-layer ReLU network trained via noisy-SGD for a univariate regularized regression problem. Our main result is that SGD with vanishingly small noise injected in the gradients is biased towards a simple solution: at convergence, the ReLU network implements a piecewise linear map of the inputs, and the number of "knot" points – i.e., points where the tangent of the ReLU network estimator changes – between two consecutive training inputs is at most three. In particular, as the number of neurons of the network grows, the SGD dynamics is captured by the solution of a gradient flow and, at convergence, the distribution of the weights approaches the unique minimizer of a related free energy, which has a Gibbs form. Our key technical contribution consists in the analysis of the estimator resulting from this minimizer: we show that its second derivative vanishes almost everywhere, except at some specific locations which represent the "knot" points. We also provide empirical evidence that knots at locations distinct from the data points might occur, as predicted by our theory.

4.1 Motivation and Outlook

We start with a quick recap of the motivation discussed in the introductory chapter of the thesis and continue with the specifics related to the current chapter analysis.

Neural networks are the key ingredient behind many recent advances in machine learning. They achieve state-of-the-art performance on various practical tasks, such as image classification [HZRS16] and synthesis [BDS19], natural language processing [VSP^{+}17] and reinforcement learning [SHM^{+}16]. However, these results would not be possible without computational advances which enabled the training of highly overparameterized models with billions of weights. Such complex networks are capable of extracting more sophisticated patterns from the data than their less parameter-heavy counterparts. Nonetheless, in the view of classical learning theory, models with a large number of parameters are prone to over-fitting [VLS11]. Contrary to the conventional statistical wisdom, overparameterization turns out to be a rather desirable property for neural networks. This was even observed in a classical paper by [Bar98], which demonstrated that in the overparameterized setting, the size of the network is less important than the magnitude of the weights. More recently, phenomena such as double descent [BHMM19, SGd+19, NKB+20] and benign overfitting [BLLT20, LZG21, BMR21] suggest that understanding the generalization properties of overparameterized models lies beyond the scope of the usual control of capacity via the size of the parameter set [NTS15].

One way to explain the generalization capability of large neural networks lies in characterizing the properties of solutions found by stochastic gradient descent (SGD). In other words, the question is whether the optimization procedure is implicitly selective, i.e., it finds the functionally simple solutions that exhibit superior generalization ability in comparison to other candidates with roughly the same value of the empirical risk. For instance, [CB20] consider shallow networks minimizing the logistic loss, and show that SGD converges to a max-margin classifier on a certain functional space endowed with the variation norm. In the machine learning literature, it has been suggested that large margin classifiers inherently exhibit better performance on unseen data [BMR21, CV95].

Constraints on the functional class of network solutions can also be imposed explicitly, e.g., via ℓ_2 regularization or by adding label noise. In some cases, it has been shown that the presence of parameter penalties or noise results in surprising implications. Depending on the regime, it biases optimization to find smooth solutions [SPD+22, JM23, SESS19] or piecewise linear functions [BGVV20, EP21]. The study by [BB18] proposes an alternative to conventional ℓ_p regularization inspired by max-affine spline operators. It enforces a neural network to learn orthogonal representations, which significantly improves the performance and does not require any modifications of the network architecture.

In this chapter, we develop a novel approach towards understanding the implicit bias of gradient descent methods applied to overparameterized neural networks. In particular, we focus on the following key questions:

> *Once stochastic gradient descent has converged, how does the distribution of the weights of the neural network look like? What functional properties of the resulting solution are induced by this stationary distribution? Can we quantitatively characterize the trade-off between the complexity of the solution and the size of the training data in the overparameterized regime?*

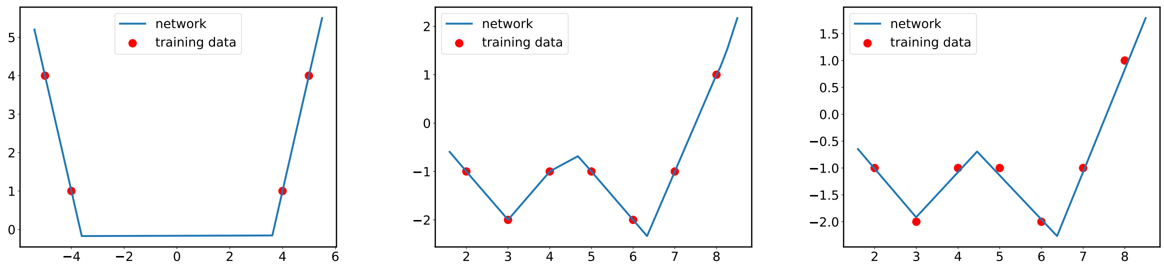
To answer these questions, we consider training a wide two-layer ReLU (rectified linear unit) network for univariate regression, and we focus on the mean-field regime [MMN18, RVE18, CB18, SS20]. In this regime, the idea is that, as the number of neurons of the network grows, the weights obtained via SGD are close to i.i.d. samples coming from the solution of a certain Wasserstein gradient flow. As a consequence, the output of the neural network approaches the following quantity:

$$y_\rho^{\sigma^*}(x) = \int \sigma^*(x, \boldsymbol{\theta})\rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}.$$

Here, x is the input, σ^* denotes the activation function, and ρ is the solution of the Wasserstein gradient flow minimizing the free energy

$$\mathcal{F}(\rho) = \frac{1}{2}\,\mathbb{E}_{(x,y)\sim\mathbb{P}}\Big\{(y - y_\rho^{\sigma^*}(x))^2\Big\} + \frac{\lambda}{2}\int \|\boldsymbol{\theta}\|_2^2\rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} + \beta^{-1}\int \rho(\boldsymbol{\theta})\log\rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}. \tag{4.1}$$

The first term corresponds to the expected squared loss (under the data distribution $\mathbb{P}$); the second term comes from the ℓ_2 regularization; the differential entropy term is linked to the



(a) (b) (c)

Figure 4.1: Example of functions learnt by a two-layer ReLU network with $N = 1000$ neurons on different training data. Solutions (a)-(b) are obtained with *no* regularization and label noise, i.e., $\lambda = 0$ and $\beta = +\infty$, while in (c) we have a sufficiently large regularization coefficient, which does not allow the network to fit the training data perfectly. Note that the piecewise linear solution exhibits tangent changes also at points different from the training data. Furthermore, the number of "knot" points may differ from the minimum required to fit the data: for instance, in (a) the minimum amount of tangent changes is 1, but the solution has two of them.

noise introduced into the SGD update, and it penalizes non-uniform solutions. The coefficient β is often referred to as *inverse temperature*. In [MMN18], it is also shown that the minimizer of the free energy, call it ρ^*, has a Gibbs form for a sufficiently regular activation function σ^*. We reviewed the connection between the dynamics of gradient descent and the solution ρ of the Wasserstein gradient flow in Chapter 2 of the current thesis. For the purposes of the this chapter analysis, we point out the differences between the more general setup discussed in Chapter 2 and the current one-dimensional regression problem in the subsequent Section 4.3.

A number of works has exploited this connection to provide a rigorous justification to various phenomena attributed to neural networks. [MMN18, MMM19] give global convergence guarantees for two-layer networks by studying the energy dissipation along the trajectory of the flow. The paper by [CB18] takes a different route and exploits a lifting property enabled by a certain type of initialization and regularization, and [JMM20] put forward an argument based on displacement convexity. [NP23] and [AOY19] tackle the multi-layer case, and, in particular, [NP23] establish convergence guarantees for a three-layer network. [FLYZ21] introduce a mean-field dynamics capturing the evolution of the features (instead of the network parameters) and show global convergence of ResNet type of architectures. In Chapter 3, we prove two properties commonly observed in practice (see e.g. [GIP$^+$18, DVSH18, KWL$^+$19]), namely dropout stability and mode connectivity, for multi-layer networks trained under the mean-field regime. [DBDFS20] consider different scalings of the step size of SGD, and identify two regimes under which different mean-field limits are obtained. [WTS$^+$19] show that the gradient flow for unregularized objectives forces the neurons of a two-layer ReLU network to concentrate around a subset of the training data points.

In this chapter, similarly to Chapter 3, we take a mean-field view to show that SGD is biased towards functionally simple solutions, namely, piecewise linear functions. Our idea is to analyze the stationary distribution ρ^* minimizing the free energy (4.1). We show that, in the low temperature regime ($\beta \to \infty$), the estimator's curvature vanishes everywhere except for a certain *cluster set*. More precisely, for each interval between two consecutive training inputs, aside for a set of small measure, the second derivative vanishes, i.e.,

$$\frac{\partial^2}{\partial x^2}\, y^{\sigma^*}_{\rho^*}(x) \to 0, \qquad \text{as } \beta \to \infty.$$

Furthermore, we provide a characterization of the cluster set and show that its measure vanishes while it concentrates around at most 3 points per interval. Ultimately, this analysis guarantees that, in the regime of decreasing temperature (corresponding to a small noise injected in the gradient updates), the solution found by SGD is piecewise linear. Our main contribution can be summarized in the following informal statement:

Theorem (Informal). *Under the low temperature regime, i.e., $\beta \to \infty$, the estimator obtained by training a two-layer ReLU network via noisy-SGD converges to a piecewise linear solution. Furthermore, the number of "knot" points – i.e., points at which distinct linear pieces connect – between two consecutive training inputs is at most 3.*

Let us remark on a few important points. In the overparameterized regime, the number of neurons N is significantly larger than the number of training samples M, i.e., $N \gg M$. The output of the two-layer ReLU network is a linear combination of N ReLU units, hence the function implemented by the network is clearly piecewise linear with $\mathcal{O}(N)$ knot points. Here, we show that the number of knot points is actually $\mathcal{O}(M) \ll \mathcal{O}(N)$. Our analysis applies for both constant $(\lambda \to \bar{\lambda} > 0)$ and vanishing $(\lambda \to 0)$ regularization, and it does not require a specific form for the initialization of the parameters of the networks (as long as some mild technical conditions are satisfied).

In a nutshell, we establish a novel technique that accurately characterizes the solution to which gradient descent methods converge, when training overparameterized two-layer ReLU networks. Our analysis unveils a behaviour which is qualitatively different from that described in recent works [WTS$^+$19, BGVV20, EP21] (see also a detailed comparison in Section 4.8): knot points are not necessarily allocated at the training points, or in a way that results in a function with the minimum number of tangent changes required to fit the data. We provide also numerical simulations to validate our findings (see Section 4.7 and Figure 4.1 above). We suggest that this novel behaviour is likely due to the difference in settings and the additional ℓ_2 regularization (including of the bias parameters).

Organization of Chapter 4. The rest of the chapter is organized as follows. In Section 4.2, we review the related work and a more detailed comparison is deferred to Section 4.8. In Section 4.3, we provide some preliminaries, including a background on the mean-field analysis in Section 4.3.1. Our main results are stated in Section 4.4 and proved in Section 4.5. In Section 4.6, we provide an example of a dataset for which the estimator found by SGD has a knot at a location different from the training inputs. We validate our findings with numerical simulations for different regression tasks in Section 4.7. We conclude and discuss some future directions in Section 4.9. Some of the technical lemmas and the corresponding proofs are deferred to Appendix B.1.

Notation. We use bold symbols for vectors $\boldsymbol{a}, \boldsymbol{b}$, and plain symbols for real numbers a, b. We use capitalized bold symbols to denote matrices, e.g., $\boldsymbol{\Theta}$. We denote the ℓ_2 norm of vectors $\boldsymbol{a}, \boldsymbol{b}$ by $\|\boldsymbol{a}\|_2, \|\boldsymbol{b}\|_2$. Given an integer N, we denote $[N] = \{1, \dots, N\}$. Given a discrete set $\mathcal{A}$, $|\mathcal{A}|$ is its cardinality. Similarly, given a Lebesgue measurable set $\mathcal{B} \subset \mathbb{R}^d$ its Lebesgue measure is given by $|\mathcal{B}|$. Given a sequence of distributions $\{\rho_n\}_{n \geq 0}$, we write $\rho_n \rightharpoonup \rho$ to denote the weak L_1 convergence of the corresponding measures. For a sequence of functions $\{f_n\}_{n \geq 0}$ we denote by $f_n \to f$ the *pointwise* convergence to a function f. Given a real number $x \in \mathbb{R}$, the closest integer that is not greater than x is defined by $\lfloor x \rfloor$.

4.2 Related Work

The line of works [WTS$^+$19, JM23] shows that, in the lazy training regime [COB19, JGH18] and for a uniform initialization, SGD converges to a cubic spline interpolating the data. Furthermore, for multivariate regression in the lazy training regime, [JM23] proved that the optimization procedure is biased towards solutions minimizing the 2-norm of the Radon transform of the fractional Laplacian. Similar results (although without the connection to the training dynamics) are obtained in [SESS19, OWSS20], which analyze the solutions with zero loss and minimum norm of the parameters. [EP21] develop a convex analytic framework to explain the bias towards simple solutions. In particular, an explicit characterization of the minimizer is provided, which implies that an optimal set of parameters yields linear spline interpolation for regression problems involving one dimensional or rank-one data. [CFW$^+$21] show that, for overparameterized models, the lower degree spherical harmonics are easier to learn. This observation comes from the fact that, in the lazy training regime, the convergence occurs faster along the directions given by the top eigenfunctions of the neural tangent kernel. Classification with linear networks on separable data is considered in [SHN$^+$18], where it is shown that gradient descent converges to the max-margin solution. This max-margin behavior is demonstrated in [CB20] for non-linear wide two-layer networks using a mean-field analysis. In particular, in the mean-field regime, optimizing the logistic loss is equivalent to finding the max-margin classifier in a certain functional space. The paper by [ZXLM20] focuses on the lazy training regime, and it shows that the optimization procedure finds a solution that fits the data perfectly and is closest to the starting point of the dynamics in terms of Euclidean distance in the parameter space. [WZBG21] characterize the directional bias of GD and SGD in the case of moderate (but annealing) learning rate.

The behavior of SGD with label noise near the zero-loss manifold is studied in [BGVV20]. Here, it is shown that the training algorithm implicitly optimizes an auxiliary objective, namely, the sum of squared norms of the gradients evaluated at each training sample. This allows the authors of [BGVV20] to show that SGD with label noise for a two-layer ReLU network with skip-connections is biased towards a piecewise linear solution. In particular, this piecewise linear solution has the minimum number of tangent changes required to fit the data. [WTS$^+$19] consider the Wasserstein gradient flow on a certain space of reduced parameters (in polar coordinates), and show that the points where the solution changes tangent are concentrated around a subset of training examples. A trade-off between the scale of the initialization and the training regime is also provided in [WTS$^+$19, SPD$^+$22]. [MBG18] prove that the gradient flow enforces the weight vectors to concentrate at a small number of directions determined by the input data. Through the lens of spline theory, [PN20b] explain that a number of best practices used in deep learning, such as weight decay and path-norm, are connected to the ReLU activation and its smooth counterparts. [NLB$^+$19] suggest a novel complexity measure for neural networks that provides a tighter generalization for the case of ReLU activation.

4.3 Preliminaries

4.3.1 Mean-field Framework

We now elaborate on a few differences between the multi-dimensional mean-field setup, i.e., $\boldsymbol{x} \in \mathbb{R}^d$, discussed in Chapter 2 and the current chapter unidimensional regression problem given the *finite* data.

We consider a regression problem for a dataset $\{(x_j, y_j)\}_{j=1}^{M}$ containing M points, here both inputs and outputs are unidimensional, i.e., $x_i \in \mathbb{R}$ and $y_i \in \mathbb{R}$. This means that in all the related definitions and equations presented in Chapter 2 one should substitute bold font vector $\boldsymbol{x}$ with the regular font scalar x. We assume that the inputs are sorted in increasing order, i.e.,

$$x_j < x_{j+1} \; \forall j \in [M-1].$$

However, it is necessary to point out that we do not tackle the case of duplicate inputs, i.e.,

$$\text{there exists } i \neq j \in [M] \text{ such that } x_i = x_j.$$

In the view of finite data size M, we have that the data distribution $\mathbb{P}$, that appears in the population risk (2.29) and a few related quantities such as potential $\Psi_\lambda(\boldsymbol{\theta}, \rho)$ in (2.33), has the following form

$$\mathbb{P} := \frac{1}{M} \sum_{j=1}^{M} \delta_{(x_j, y_j)},$$

where, $\delta_{(a,b)}$ stands for a delta distribution centered at $(a, b) \in \mathbb{R}^2$. This means that the quantities that involve the expectation over $\mathbb{P}$ can now be rewritten using the summation over the training data. For example, the population risk (2.29) will take the form

$$\frac{1}{M} \sum_{i=1}^{M} \left[\left(\hat{y}_N(x_i, \boldsymbol{\Theta}) - y_i\right)^2 \right] + \frac{\lambda}{N} \sum_{i=1}^{N} \|\boldsymbol{\theta}_i\|_2^2.$$

The summation form is later used for the Gibbs minimizers (4.4) and (4.5), and population risks that correspond to different approximations of the activation σ^* that we discuss in the subsequent Section (4.3.2).

We also point out the generic regularity conditions (2.31) imposed on the step size s_k scaling $\xi(t)$ are assumed to hold throughout the current chapter. In this view, we permit ourselves a leisure to not explicitly mention this mild condition in the statements of the formal results.

4.3.2 Approximation of the ReLU Activation

Let us elaborate on the properties which σ^* should satisfy so that the results described in Chapter 2 hold. First, the distributional dynamic (2.33) is known to be well-defined for a smooth and bounded potential Ψ_λ. In particular, it suffices to choose a bounded, Lipschitz σ^* with Lipschitz gradient, see assumptions A2-A3 in [MMN18]. Furthermore, the minimizer of the free energy (2.36) exists and has a Gibbs form even for non-smooth potentials and, in particular, it suffices that σ^* is bounded and Lipschitz (this allows the first derivative to be discontinuous), see Lemmas 10.2-10.4 in [MMN18].

In the case of a ReLU activation, the corresponding σ^* has the following form

$$\sigma^*(x, \boldsymbol{\theta}) = a(wx + b)_+ = a \max\{0, wx + b\}, \quad \boldsymbol{\theta} = (a, w, b) \in \mathbb{R}^3,$$

which does not satisfy some of the aforementioned conditions. The first salient problem is the lack of continuity of the derivative at zero. This issue can be dealt with by considering a soft-plus activation with scale τ:

$$(x)_\tau := \frac{\log(1 + e^{\tau x})}{\tau}.$$

Notice that, as τ grows large, we have that $(\cdot)_\tau \to (\cdot)_+$. Another issue is that the function $\sigma^*(x, \boldsymbol{\theta})$ is not Lipschitz in the parameters $\boldsymbol{\theta}$, and it is unbounded. This problem can be solved by an appropriate truncation applied to the parameter a of the activation. The truncation should be Lipschitz and smooth for the dynamics to be well-defined.

In this view, we now provide the details of the approximation of the ReLU activation. For a parameter $v \in \mathbb{R}$, we denote by v^m its m-truncation defined as

$$v^m := \mathbb{1}_{\{|v|>m\}} \cdot m \cdot \operatorname{sign}(v) + \mathbb{1}_{\{|v|\le m\}} \cdot v.$$

Notice that the function $f(v) = v^m$ is Lipschitz continuous and bounded. For a parameter $v \in \mathbb{R}$, we denote by $v^{\tau,m}$ its τ-smooth m-truncation defined as follows: $v^{\tau,m}$ converges pointwise to v^m as $\tau \to \infty$, $v^{\tau,m} = v$ inside the ball $\{v \in \mathbb{R} : |v| < m - \frac{1}{\tau}\}$, and the map $v \mapsto v^{\tau,m}$ is odd and belongs to $C^4(\mathbb{R})$. For a visualization of v^m and $v^{\tau,m}$, see Figure 4.2a.

We define the smooth m-truncation $(\cdot)_+^m$ of the ReLU activation as

$$(x)_+^m := \mathbb{1}_{\{x\le m^2\}}(x)_+ + \mathbb{1}_{\{x>m^2\}}\phi_m(x),$$

where ϕ_m is chosen so that the following holds: $(x)_+^m \in C^4(\mathbb{R})$, $(x)_+^m \le (x)_+$ for all $x \in \mathbb{R}$, and $|\phi_m''(x)| \le \frac{1}{m^2}$ for $x > m$. Note that these conditions imply that $\phi_m(m^2) = m^2$ and $\phi_m'(m^2) = 1$. Furthermore, in order to enforce the bound on ϕ_m'', we pick ϕ_m so that $\lim_{x\to+\infty} \phi_m(x) = 2m^2$, and $\lim_{x\to+\infty} \phi_m'(x) = \lim_{x\to+\infty} \phi_m''(x) = 0$. For a visualization of $(\cdot)_+^m$, see Figure 4.2b.

Finally, we define the smooth m-truncation $(\cdot)_\tau^m$ of the softplus activation as

$$(x)_\tau^m := \mathbb{1}_{\{x\le x_m\}}(x)_\tau + \mathbb{1}_{\{x>x_m\}}\phi_{\tau,m}(x), \tag{4.2}$$

where $x_m \in \mathbb{R}$ is such that $(x_m)_\tau = m^2$. As in the truncation of ReLU, we choose $\phi_{\tau,m}$ so that $(x)_\tau^m \in C^4(\mathbb{R})$ and $|\phi_{\tau,m}''(x)| \le \frac{1}{m^2}$ for $x > x_m$. Furthermore, we require that $(\cdot)_\tau^m$ converges pointwise to $(\cdot)_+^m$ as $\tau \to \infty$ (which we can guarantee since $(\cdot)_\tau \to (\cdot)_+$, as $\tau \to \infty$). To enforce these conditions, we pick $\phi_{\tau,m}$ so that $\phi_{\tau,m}(x_m) = m^2$, $\phi_{\tau,m}'(x_m) = (x)_\tau'\big|_{x=x_m}$, $\lim_{x\to+\infty} \phi_{\tau,m}(x) = 2m^2$, and $\lim_{x\to+\infty} \phi_{\tau,m}'(x) = \lim_{x\to+\infty} \phi_{\tau,m}''(x) = 0$. For a visualization of $(\cdot)_\tau^m$, see Figure 4.2c.

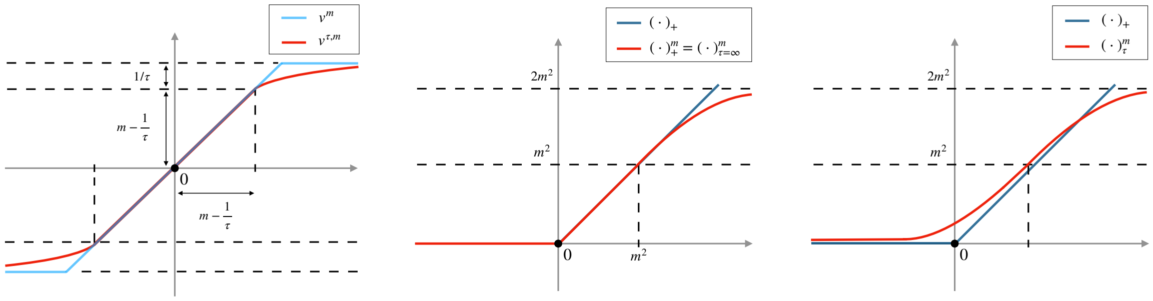
Notice that, for $\tau \ge 1$, the soft-plus activation can be sandwiched as follows:

$$(x)_+ - \frac{1}{\tau} \le (x)_\tau \le (x)_+ + \frac{1}{\tau}.$$

In order to establish the continuity of a certain limit and smoothness properties, we also pick $\phi_{\tau,m}$ such that the smooth m-truncation of soft-plus activation satisfies a similar bound:

$$(x)_+ - \frac{1}{\tau} \le (x)_\tau^m \le (x)_+ + \frac{1}{\tau}. \tag{4.3}$$

At this point, we remark that the activation $(\boldsymbol{\theta}, x) \mapsto a^{\tau,m}(w^{\tau,m}x + b)_\tau^m$ satisfies all the conditions necessary for the results of Section 4.3.1 to hold. In what follows, we will also use the activation $(\boldsymbol{\theta}, x) \mapsto a^m(w^m x + b)_\tau^m$ as an auxiliary object. This map is not smooth, but it satisfies all the assumptions required for the existence of a free energy minimizer $\rho_{\sigma^*}^*$. We also note that the truncation of the parameter w might seem unnatural (we are truncating the ReLU activation anyway), but it simplifies our analysis. In particular, it allows us to establish



(a) v^m and $v^{\tau,m}$ (b) $(\cdot)_+$ and $(\cdot)_+^m$ (c) $(\cdot)_+$ and $(\cdot)_\tau^m$

Figure 4.2: Visualization of the functions involved in the approximation of the ReLU activation.

a connection between the derivatives (w.r.t. the input x) of the predictor implemented by the solution of the flow (2.33) and the same quantity evaluated on the minimizer, as t grows large.

We will use the following notation for the values of the risks corresponding to different activations

$$R_i^{\tau,m}(\rho) := -\frac{1}{M}\left(y_i - \int a^{\tau,m}(w^{\tau,m}x_i + b)_\tau^m \rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\right), \quad R^{\tau,m}(\rho) := M\sum_{i=1}^{M}\left(R_i^{\tau,m}(\rho)\right)^2,$$
$$R_i^{m}(\rho) := -\frac{1}{M}\left(y_i - \int a^{m}(w^{m}x_i + b)_+^m \rho(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\right), \quad R^{m}(\rho) := M\sum_{i=1}^{M}\left(R_i^{m}(\rho)\right)^2,$$

and for the related free-energies

$$\mathcal{F}^{\tau,m}(\rho) := \frac{1}{2}R^{\tau,m}(\rho) + \frac{\lambda}{2}M(\rho) - \beta^{-1}H(\rho),$$
$$\mathcal{F}^{m}(\rho) := \frac{1}{2}R^{m}(\rho) + \frac{\lambda}{2}M(\rho) - \beta^{-1}H(\rho).$$

Here, $R_i^{\tau,m}$ and R_i^m represent the rescaled error on the i-th training sample, and $R^{\tau,m}$ and R^m are the standard expected square losses. In this way, we can write the Gibbs minimizers in a compact form, namely,

$$\rho_{\tau,m}^*(\boldsymbol{\theta}) = Z_{\tau,m}^{-1}(\beta,\lambda)\exp\left\{-\beta\left[\sum_{i=1}^{M} R_i^{\tau,m}(\rho_{\tau,m}^*)\cdot a^{\tau,m}(w^{\tau,m}x_i + b)_\tau^m + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\}, \tag{4.4}$$

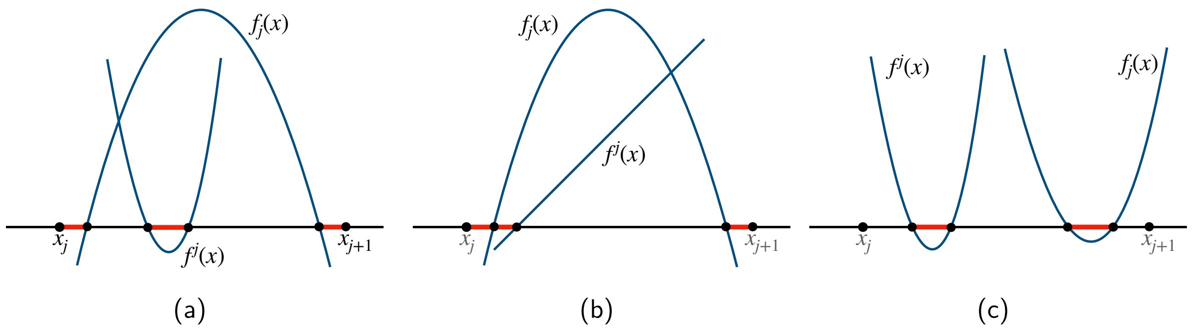
$$\rho_{m}^*(\boldsymbol{\theta}) = Z_{m}^{-1}(\beta,\lambda)\exp\left\{-\beta\left[\sum_{i=1}^{M} R_i^{m}(\rho_{m}^*)\cdot a^{m}(w^{m}x_i + b)_+^m + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\}, \tag{4.5}$$

where $Z_{\tau,m}(\beta,\lambda)$ and $Z_m(\beta,\lambda)$ denote the partition functions.

4.4 Main Results

Before presenting the main results, let us introduce the notion of a *cluster set*. This set allows us to identify the locations of the knot points of an estimator function that is implemented by the neural network. In particular, we consider the second derivative of the predictor evaluated at the Gibbs distribution with activation $(\boldsymbol{\theta}, x) \mapsto a^{\tau,m}(w^{\tau,m}x + b)_\tau^m$, for large τ, i.e.,

$$\lim_{\tau\to\infty}\frac{\partial^2}{\partial x^2}\int a^{\tau,m}(w^{\tau,m}x + b)_\tau^m \rho_{\tau,m}^*(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}. \tag{4.6}$$



(a) (b) (c)

Figure 4.3: Three different configurations of the polynomials $f^j(x)$ and $f_j(x)$, together with the corresponding cluster set. The dark blue curves show the shape of polynomials, and the red bold intervals indicate the set on which polynomials attain non-positive value.

Then, the cluster set is associated to the inputs on which the quantity (4.6) might grow unbounded in absolute value, in the low temperature regime ($\beta^{-1} \to 0$). Intuitively, this indicates that on some points of the cluster set, the tangent of the predictor changes abruptly, resulting in "knots". We denote the cluster set by $\Omega(m, \beta, \lambda)$, and we define it below.

Let $\mathcal{I}$ be the set of prediction intervals, i.e.,

$$\mathcal{I} = \Big\{[x_0 := -L, x_1], [x_1, x_2], \cdots, [x_{M-1}, x_M], [x_M, x_{M+1} := L]\Big\},$$

where $L > \max\{|x_1|, \cdots, |x_M|\}$ is any fixed positive constant independent of $(\tau, m, \beta, \lambda)$. For each $I_j := [x_j, x_{j+1}] \in \mathcal{I}$, the intersection of the cluster set with the prediction interval I_j is denoted by $\overline{\Omega}_j(m, \beta, \lambda)$, i.e.,

$$\overline{\Omega}_j(m, \beta, \lambda) = \Omega(m, \beta, \lambda) \cap I_j. \tag{4.7}$$

Thus, in order to define the cluster set $\Omega(m, \beta, \lambda)$, it suffices to give the definition of $\overline{\Omega}_j(m, \beta, \lambda)$. To do so, consider the second-degree polynomials $f^j(x)$ and $f_j(x)$ given by

$$\begin{aligned} f^j(x) &:= 1 + x^2 - (A^j x - B^j)^2, \\ f_j(x) &:= 1 + x^2 - (A_j x - B_j)^2, \end{aligned} \tag{4.8}$$

with coefficients

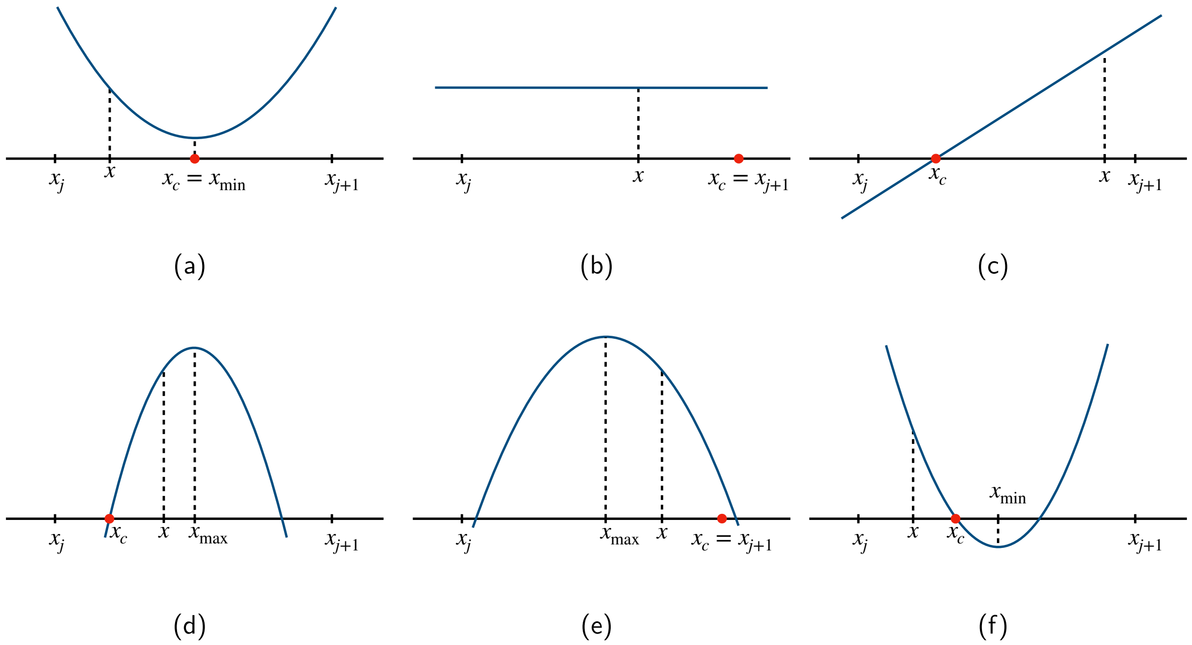
$$\begin{aligned} A^j &:= \frac{1}{\lambda} \sum_{i=j+1}^{M} R_i^m(\rho_m^*), \quad A_j := \frac{1}{\lambda} \sum_{i=1}^{j} R_i^m(\rho_m^*), \\ B^j &:= \frac{1}{\lambda} \sum_{i=j+1}^{M} R_i^m(\rho_m^*) x_i, \quad B_j := \frac{1}{\lambda} \sum_{i=1}^{j} R_i^m(\rho_m^*) x_i. \end{aligned} \tag{4.9}$$

Here, if the summation set is empty (e.g., for A_0), the corresponding coefficient is equal to zero. Then, the set $\overline{\Omega}_j(m, \beta, \lambda)$ is defined as the union of the non-positive sets of the second-degree polynomials $f^j(x)$ and $f_j(x)$:

$$\overline{\Omega}_j(m, \beta, \lambda) = \Omega^j(m, \beta, \lambda) \cup \Omega_j(m, \beta, \lambda), \tag{4.10}$$

where

$$\begin{aligned} \Omega^j(m, \beta, \lambda) &:= \{x \in I_j : f^j(x) \le 0\}, \\ \Omega_j(m, \beta, \lambda) &:= \{x \in I_j : f_j(x) \le 0\}. \end{aligned} \tag{4.11}$$



(a) (b) (c)

(d) (e) (f)

Figure 4.4: Representation of the critical point x_c for different configurations of the polynomial f^j and evaluation point x. The red dot indicates the location of the critical point. The dashed line indicates the value of f^j attained at the corresponding point. The dark blue curve shows the shape of the polynomial f^j.

We now provide an informal explanation on how the non-positive sets of the second-degree polynomials $f^j(x)$ and $f_j(x)$ come into play. A central object of interest in our analysis is the second derivative of the estimator implemented by the neural network, and our strategy is to bound its magnitude by a particular Gaussian-like integral. This integral does not diverge as long as the corresponding covariance matrix is non-degenerate, i.e., it has strictly positive eigenvalues. In this view, the non-positive sets of the polynomials $f^j(x)$ and $f_j(x)$ precisely characterize the inputs x for which this covariance matrix is degenerate. Hence, for such inputs x, this upper bound on the second derivative of the estimator diverges, which implies that the predictor may have a "knot".

Since $f^j(x)$ and $f_j(x)$ are second-degree polynomials, the set $\overline{\Omega}_j(m, \beta, \lambda)$ can be always written as the union of at most 3 intervals. Moreover, $\overline{\Omega}_j(m, \beta, \lambda)$ depends only on the errors of the estimator at the training points and on the penalty parameter λ. Thus, if one has access to the value of the errors at each training point for the optimal estimator, i.e., $R_i^m(\rho_m^*)$, an explicit expression for the cluster set can be readily obtained. Figure 4.3 shows three different configurations of the polynomials $f^j(x)$ and $f_j(x)$, together with the corresponding cluster set.

The size of the set $\overline{\Omega}_j(m, \beta, \lambda)$ can be controlled explicitly as a function of the parameters (m, β, λ). More formally, in Lemma 4.5.3, we show that the Lebesgue measure of $\overline{\Omega}_j(m, \beta, \lambda)$ can be upper bounded as

$$|\overline{\Omega}_j(m, \beta, \lambda)| \leq \frac{e^{C\beta}}{m^2}, \tag{4.12}$$

where $C > 0$ denotes a numerical constant independent of $(\tau, m, \beta, \lambda)$ and we have made the following assumption:

B1. $\tau \geq 1$, $\beta \geq \max\left\{C_1, \frac{1}{\lambda}, \frac{1}{\lambda}\log\frac{1}{\lambda}\right\}$, $m > C_2$ and $\lambda < C_3$ for some numerical constants

$C_1, C_2, C_3 > 0$.

In particular, (4.12) implies that the cluster set vanishes as $\beta \to \infty$ and $m = e^{\Theta(\beta)}$. Therefore, as $\overline{\Omega}_j(m, \beta, \lambda)$ is the union of at most 3 intervals, the cluster set concentrates on at most 3 points per prediction interval.

We note that our use of **B1** throughout the sequel is with the flexibility of C_1, C_2, and C_3 in mind; we are interested in the behavior as m and β grow large, so we permit liberty in the determination of the constants implying the formal statements we intend to show.

A key step of our analysis (cf. Theorem 3) consists in showing that, outside the cluster set, the absolute value of the second derivative vanishes. Our bound on this absolute value is connected to the speed of decay to zero of the polynomials $f^j(x)$ and $f_j(x)$, as the input x approaches the cluster set. In order to establish a quantitative bound for such a decay, we introduce an auxiliary quantity, namely, a *critical point*, that is associated to each input point outside of the cluster set. Given the polynomial $f^j(\cdot)$ and the input $x \in I_j \setminus \Omega^j(m, \beta, \lambda)$, the critical point x_c associated to x is defined below.

Definition 4.4.1 (Critical point). *If $f_j(\tilde{x}) = 0$ has no solutions for $\tilde{x} \in \mathbb{R}$, then the critical point x_c associated to x and $I_j \setminus \Omega^j(m, \beta, \lambda)$ is defined to be the minimizer of $f_j(\cdot)$ on I_j, i.e., $x_c = \arg\min_{\tilde{x} \in I_j} f_j(\tilde{x})$. In case of multiple minimizers, e.g., $(a, b) = (1, 0)$, we set $x_c = x_{j+1}$. If $f_j(\tilde{x}) = 0$ has at least one solution for $\tilde{x} \in \mathbb{R}$, then we let x_r be the root of f_j (in $\mathbb{R}$ and not necessarily in the segment I_j) which is the closest in Euclidean distance to x, and we define the* critical point *x_c to be the closest point to x_r in I_j, i.e., $x_c = x_r$ if $x_r \in I_j$ and x_c is one of the two extremes of the interval otherwise.*

Figure 4.4 provides a visualization of the critical point associated to x for several configurations of f^j. For the polynomial $f_j(\cdot)$ and an input $x \in I_j \setminus \Omega_j(m, \beta, \lambda)$, the critical point $\bar{x}_c$ is defined in a similar fashion. In this view, we show in Lemma 4.5.5 that the following lower bounds on f^j, f_j hold for $x \in I_j \setminus \overline{\Omega}(m, \beta, \lambda)$,

$$C^j(x) := \gamma_1 (x - x_c)^2 + \gamma_2 \leq f^j(x), \quad C_j(x) := \gamma_3 (x - \bar{x}_c)^2 + \gamma_4 \leq f_j(x). \tag{4.13}$$

The coefficients $\gamma_1, \gamma_2, \gamma_3, \gamma_4 > 0$ satisfy the following condition: either $\gamma_1 > \varepsilon$ or $\gamma_2 > \varepsilon$, and either $\gamma_3 > \varepsilon$ or $\gamma_4 > \varepsilon$, where $\varepsilon > 0$ is a numerical constant independent of the choice of (m, β, λ).

At this point, we are ready to state our upper bound on the second derivative outside the cluster set.

Theorem 3 (Vanishing curvature). *Assume that condition **B1** is satisfied and that $m > e^{K_1 \beta}$ for some numerical constant $K_1 > 0$ independent of $(\tau, m, \beta, \lambda)$. Then, for each $x \in I_j \setminus \overline{\Omega}_j(m, \beta, \lambda)$, the following upper bound on the second derivative holds*

$$\lim_{\tau \to +\infty} \left| \frac{\partial^2}{\partial x^2} \int a^{\tau, m} (w^{\tau, m} x + b)_\tau^m \rho^*_{\tau, m}(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \right| \leq \mathcal{O}\left(\frac{1}{m\lambda} + \frac{1}{\beta \lambda^{7/4} (\bar{C}^j(x))^2} \right), \tag{4.14}$$

where the coefficient $\bar{C}^j(x)$ is defined as

$$\bar{C}^j(x) = \min\left\{ C^j(x), C_j(x), 1 \right\}, \tag{4.15}$$

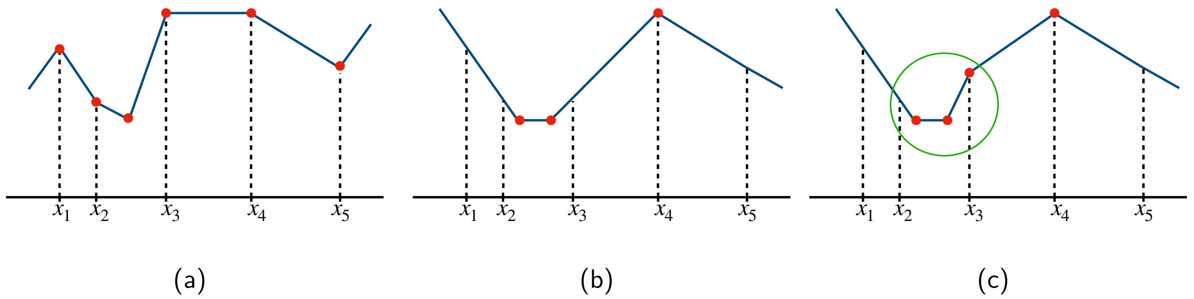


Figure 4.5: Three examples of piecewise linear functions that fit the data with zero squared error. Dashed black line indicates the y value for each training input. Red dots are located at points where the function changes its tangent. (a) and (b) illustrates two admissible piecewise linear solutions, while (c) is not admissible due to the location of break points on interval $[x_2, x_3]$.

with $C^j(x)$ and $C_j(x)$ given by (4.13). Furthermore, the following upper-bound on the size of the cluster set holds

$$|\Omega(m, \beta, \lambda)| \leq \frac{K_2}{m}, \tag{4.16}$$

for some numerical constant $K_2 > 0$ independent of $(\tau, m, \beta, \lambda)$.

Some remarks are in order. First, the inequality (4.14) shows that, in the low temperature regime, the curvature vanishes outside the cluster set, and it also provides a decay rate. Second, we will upper bound the measure of the cluster set as in (4.12), thus the condition $m > e^{K_1 \beta}$ ensures that the upper bound (4.16) holds. Finally, the presence of the coefficient $\bar{C}^j(x)$ is due to the fact that the second derivative can grow unbounded for points approaching the cluster set. Let us highlight that this growth is solely dictated by the distance to the cluster set, and it does not depend on (m, β, λ). In fact, (4.13) holds, where one of the coefficients in $\{\gamma_1, \gamma_2\}$ and in $\{\gamma_3, \gamma_4\}$ is lower bounded by a strictly positive constant independent of (m, β, λ).

From Theorem 3, we conclude that, as $m\lambda \to \infty$ and $\beta\lambda^{7/4} \to \infty$, the second derivative vanishes for all $x \in I_j \setminus \overline{\Omega}_j(m, \beta, \lambda)$. Furthermore, for $m > e^{C\beta}$ and $\beta \to \infty$, the cluster set concentrates on at most 3 points per interval. Therefore, the estimator $\int a^{\tau,m}(w^{\tau,m}x + b)^m_\tau \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}$ is piecewise linear with "knot" points given by the cluster set (cf. Theorem 4). To formalize this result, we define the notion of an *admissible piecewise linear solution*.

Definition 4.4.2 (Admissible piecewise linear solution)**.** *Given a set of prediction intervals $\mathcal{I}$, a function $f : \mathbb{R} \to \mathbb{R}$ is an admissible piecewise linear solution if f is continuous, piecewise linear and has at most 3 knot points (i.e., the points where a change of tangent occurs) per prediction interval $I_j \in \mathcal{I}$. Moreover, the only configuration possible for 3 knots to occur is the following: two knots are located strictly at the end points of the interval, and the remaining point lies strictly in the interior of the interval.*

Figure 4.5 provides some examples of piecewise linear solutions: (a) and (b) are admissible (in the sense of Definition 4.4.2), while (c) is not admissible, since it has two knots in the interior of the prediction interval and one located at the right endpoint. As mentioned before, the location of the knot points is associated with the limiting behaviour of the corresponding polynomials $f^j(x)$ and $f_j(x)$. For instance, consider the prediction interval $[x_2, x_3] \in \mathcal{I}$. Then, the configuration of Figure 4.5a corresponds to the case described in Figure 4.3a. In fact, f^j has a negative leading coefficient, and its roots are converging to the end points of the interval. Moreover, f_j has positive curvature and the minimizer is located inside the interval. The same parallel can be drawn between Figure 4.5b and Figure 4.3c. Furthermore, one can verify that the situation described in Figure 4.5c cannot be achieved for any configuration of $f^j(x)$ and $f_j(x)$.

We are now ready to state our result concerning the structure of the function obtained from the Gibbs distribution $\rho^*_{\tau,m}$.

Theorem 4 (Free energy minimizer solution is increasingly more piecewise linear)**.** *Assume that condition **B1** is satisfied and that $m > e^{K_1\beta}$, where $K_1 > 0$ is a constant independent of $(\tau, m, \beta, \lambda)$. Then, given a set of prediction intervals $\mathcal{I}$, there exists a family of admissible piecewise linear solutions $\{f_{m,\beta,\lambda}\}$ as per Definition 4.4.2, such that, for any $I \in \mathcal{I}$ and $x \in I$, the following convergence result holds*

$$\lim_{\beta\lambda^{7/4}\to+\infty} \lim_{\tau\to+\infty} \left| f_{m,\beta,\lambda}(x) - \int a^{\tau,m}(w^{\tau,m}x + b)^m_\tau \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \right| = 0.$$

In words, Theorem 4 means that the solution resulting from the minimization of the free energy (2.36) approaches a piecewise linear function, as the noise vanishes. Let us highlight that our result tackles both the regularized case in which λ approaches a fixed positive constant and the un-regularized one in which λ vanishes (as long as its vanishing rate is sufficiently slow to ensure that $\beta\lambda^{7/4} \to \infty$). We also note that that the family $\{f_{m,\beta,\lambda}\}$ is well-behaved, i.e., on each linear region the function $f_{m,\beta,\lambda}$ has the following representation: $f_{m,\beta,\lambda} = ux + v$ for some $u, v \in \mathbb{R}$, and the coefficients $|u|, |v|$ are uniformly bounded in (m, β, λ).

The proof of Theorem 4 crucially relies on the fact that the second moment of $\rho^*_{\tau,m}$ is uniformly bounded along the sequence $\beta\lambda^{7/4} \to \infty$. In fact, the uniform bound on the second moment implies that the *first* derivatives of the predictors w.r.t. the input are uniformly bounded (even for points *inside* the cluster set), and therefore the sequence of predictors is equi-Lipschitz. This, in particular, allows us to show that the limit is well-behaved, as function changes can be controlled via Lipschitz bounds.

Let us clarify that Theorem 4 does not establish the uniqueness of the limit in (m, β, λ), i.e., that the limiting piecewise linear function is the same regardless of the subsequence. Our numerical results reported in Figures 4.1, 4.6b, 4.7 and 4.8 suggest that the limit is unique. However, a typical line of argument (see e.g. [JKO98]) would require the lower-semicontinuity of the free energy (which does not hold for $m = \infty$). Furthermore, even the uniqueness of the minimizer for $\beta = \infty$ remains unclear in our setup. Nevertheless, let us point out that the sequence $\{\rho^*_{\tau,m}\}$ is tight, since the second moments are uniformly bounded by Lemma B.1.6, and Proposition 2.3 in [HRŠS21] suggests that at least the cluster points of the sequence $\{\rho^*_{\tau,m}\}$ as $\beta \to \infty$ coincide with the set of minimizers of the limiting objective ($\beta = \infty$). Another piece of evidence comes from the fact that the annealed dynamics converges to the minimizers of the noiseless objective [Chi22]. We leave for future work the resolution of these issues.

We remark that providing a quantitative bound on the parameter τ appears to be challenging. The current analysis relies on a dominated convergence argument which does not lead to an explicit convergence rate. Obtaining such a rate requires understanding the trade-off between the terms in the free energy (2.36) for varying τ, and it is also left for future work.

Finally, by combining Theorem 4 with the mean-field analysis in [MMN18], we obtain the desired result on finite-width networks trained via noisy SGD in the low temperature regime.

Corollary 4.4.3 (Noisy SGD solution is increasingly more piecewise linear)**.** *Assume that condition **B1** holds and that $m > e^{K_1 \beta}$, where $K_1 > 0$ is a constant independent of $(\tau, m, \beta, \lambda)$. Let ρ_0 be absolutely continuous and K_0 sub-Gaussian, where $K_0 > 0$ is some numerical constant. Assume also that $M(\rho_0) < \infty$ and $H(\rho_0) > -\infty$. Let $\sigma^*(x, \boldsymbol{\theta}) = a^{\tau,m}(w^{\tau,m}x + b)_\tau^m$ be the activation function, and let $\boldsymbol{\theta}^k$ be obtained by running $k = \lfloor t/\varepsilon \rfloor$ steps of the noisy SGD algorithm (2.30) with data $(\tilde{x}_k, \tilde{y}_k)_{k \geq 0} \overset{\text{i.i.d.}}{\sim} \mathbb{P}$ and initialization ρ_0. Then, given a set of prediction intervals $\mathcal{I}$, there exists a family of admissible piecewise linear solutions $\{f_{m,\beta,\lambda}\}$ as per Definition 4.4.2, such that, for any $I \in \mathcal{I}$ and $x \in I$, the following convergence result holds almost surely:*

$$\lim_{\beta\lambda^{7/4} \to +\infty} \lim_{\tau \to +\infty} \lim_{t \to +\infty} \lim_{\substack{\varepsilon \to 0 \\ N \to \infty}} \left| f_{m,\beta,\lambda}(x) - \frac{1}{N} \sum_{i=1}^{N} \sigma^*(x, \boldsymbol{\theta}_i^k) \right| = 0,$$

where the limit in N, ε is taken along any subsequence $\{(N, \varepsilon = \varepsilon_N)\}$ with $N/\log(N/\varepsilon_N) \to \infty$ and $\varepsilon_N \log(N/\varepsilon_N) \to 0$.

In words, Corollary 4.4.3 means that, at convergence, the estimator implemented by a wide two-layer ReLU network approaches a piecewise linear function, in the regime of vanishingly small noise. In fact, as $\tau, m \to \infty$, the activation function $\sigma^*(x, \boldsymbol{\theta}) = a^{\tau,m}(w^{\tau,m}x + b)_\tau^m$ converges pointwise to the ReLU activation $a(wx + b)_+$. We also remark that our result holds for any initialization of the weights of the network, as long as some mild technical conditions are fulfilled (absolute continuity, sub-Gaussian tails, finite second moment and entropy).

Let us clarify some technical aspects of the statement of Corollary 4.4.3. The result holds for a particular sequence of minimizers, since some of the limits ($t \to \infty$, $(N, \varepsilon^{-1}) \to \infty$, and $\beta \to \infty$) are not interchangeable. Furthermore, it appears to be difficult to prove the same statement directly for the noiseless case ($\beta = \infty$). We also point out that the stochasticity of the gradient descent algorithm does not play a role in our analysis, since its impact is seen to be inconsequential by the usual concentration argument [MMN18] when passing to its non-stochastic counterpart.

As concerns the limit in t, describing the dependence of the mixing time of the diffusion dynamics (2.33) on the temperature parameter β is a cumbersome task. In particular, [GBEK04] show that an exponentially bad dependence could occur if the target function has multiple small risk regions. However, some recent studies show an exponentially fast convergence of the noisy dynamics under some reasonable but particular conditions on the objective landscape [Chi22, NWS22].

As concerns the limit in (N, ε), the analyses in [MMN18, MMM19] lead to an upper bound on the error term that, with probability at least $1 - e^{-z^2}$, is given by

$$Ce^{Ct}\sqrt{1/N \vee \varepsilon} \cdot \left[\sqrt{1 + \log(N(t/\varepsilon \vee 1))} + z\right], \tag{4.17}$$

where $a \vee b$ denotes the maximum between a and b. The exponential dependence of (4.17) in the time t of the dynamics is a common drawback of existing mean-field analyses, and improving it is an open problem which lies beyond the scope of this work. Let us conclude by mentioning that the numerical results presented in Section 4.7 suggest that, in practical settings, the convergence to the limit occurs rather quickly in the various parameters.

4.5 Proof of the Main Results

4.5.1 Roadmap of the Argument

We start by providing an informal outline of the proof for the main statements. In Section 4.5.2, we show that, in the low temperature regime, the curvature of the predictor evaluated at the Gibbs distribution $\rho^*_{\tau,m}$ vanishes everywhere except at a small neighbourhood of at most three points per prediction interval $I_j \in \mathcal{I}$ (Theorem 3). This is done in a few steps. First, in Lemma 4.5.1, we show that, as $\tau \to \infty$, the density $\rho^*_{\tau,m}$ acts similarly to a delta distribution supported on the lower-dimensional linear subspace $\{b \in \mathbb{R} : b = -w^m x\}$, namely,

$$\lim_{\tau\to\infty} \frac{\partial^2}{\partial x^2} \int a^{\tau,m}(w^{\tau,m}x + b)^m_\tau \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \approx \int a^m (w^m)^2 \rho^*_m(a, w, -w^m x)\mathrm{d}a\mathrm{d}w. \tag{4.18}$$

To do so, in Lemma B.1.4 we prove that, as $\tau \to \infty$, the sequence $\rho^*_{\tau,m}(\boldsymbol{\theta})$ of minimizers of the free energy $\mathcal{F}^{\tau,m}$ converges pointwise for all $\boldsymbol{\theta}$ to a minimizer $\rho^*_m(\boldsymbol{\theta})$ of the free energy $\mathcal{F}^m$ with truncated ReLU activation. Then, a dominated convergence argument allows us to obtain (4.18). Next, in Lemma 4.5.7 we show that, as $\beta \to \infty$, the absolute value of the integral

$$\int a^m (w^m)^2 \rho^*_m(a, w, -w^m x)\mathrm{d}a\mathrm{d}w \tag{4.19}$$

can be made arbitrary small for all x except those in the cluster set. The idea is that the absolute value of (4.19) can be bounded by a certain Gaussian integral, and the corresponding covariance matrix is well-defined everywhere except in the cluster set (see Lemmas 4.5.4 and 4.5.5). The definition of the cluster set (see (4.7)-(4.11)) together with the fact that the partition function of ρ^*_m is uniformly bounded in m (see Lemma 4.5.2) allows us to show that the cluster set concentrates on at most three points per interval as $\beta \to \infty$.

In Section 4.5.3, we show that the predictor evaluated at the Gibbs distribution $\rho^*_{\tau,m}$ can be approximated arbitrarily well by an admissible piecewise linear solution (Theorem 4). First, via a Taylor argument, since the curvature vanishes, the estimator can be approximated by a linear function on each interval of $\mathcal{I} \setminus \Omega(m, \beta, \lambda)$. Since the cluster set vanishes concentrating on at most three points per prediction interval, the predictor converges to an admissible piecewise linear solution. However, there is one technical subtlety to consider before reaching this conclusion. Namely, we must consider the possibility that the sequence of predictors experiences unbounded oscillations inside the cluster set, which might ultimately result in a discontinuous limit. Fortunately, this scenario is ruled out because the sequence $\rho^*_{\tau,m}$ has uniformly bounded second moments. This fact in conjunction with the structure of the *first* derivative of the predictor yields the conclusion that the sequence of predictors is equi-Lipschitz, and therefore the limit is well-behaved.

Finally, the proof of Corollary 4.4.3 follows from similar arguments together with the application of the result established in [MMN18]. More specifically, first, the truncation of the parameter w ensures that, as $t \to \infty$, the curvature of the predictor evaluated on the solution ρ_t of the flow (2.33) converges pointwise in x to the corresponding evaluation on the Gibbs distribution $\rho^*_{\tau,m}$. Next, following [MMN18], we couple the weights obtained after $\lfloor t/\varepsilon \rfloor$ steps of the SGD iteration (2.30) with N i.i.d. particles with distribution ρ_t, thus obtaining that the curvature of the SGD predictor converges to the curvature of the flow predictor. By using this coupling again, together with the fact that along the trajectory of the flow $M(\rho_t) < C$ (see [MMN18] or [JKO98]), we obtain a uniform bound on the second moment of the empirical distribution $\hat{\rho}^N_{\lfloor t/\varepsilon \rfloor}$ of the SGD weights. The final result then follows from the same Lipschitz argument described above.

4.5.2 Proof of Theorem 3

Let us start with the proof of the vanishing curvature phenomenon. The quantity

$$\frac{\partial^2}{\partial x^2}\int a^{\tau,m}(w^{\tau,m}x+b)_{\tau}^{m}\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \tag{4.20}$$

is hard to analyze directly due to the presence of the τ-smoothing in the soft-plus activation. However, the structure of the activation $(\cdot)_{\tau}^{m}$ alongside with the pointwise convergence of the minimizers $\rho_{\tau,m}^{*}$ to ρ_{m}^{*} (cf. Lemma B.1.4) allows us to infer the properties of (4.20) through the analysis of the auxiliary object:

$$\int a^{m}(w^{m})^{2}\rho_{m}^{*}(a,w,-w^{m}x)\mathrm{d}a\mathrm{d}w. \tag{4.21}$$

Formally, we show that the approximation result below holds.

Lemma 4.5.1 (Convergence to delta). *Assume that condition **B1** holds. Let $\rho_{\tau,m}^{*}$ and ρ_{m}^{*} be the minimizers of the free energy for truncated softplus and ReLU activations, respectively, as defined in (4.4)-(4.5). Then,*

$$\lim_{\tau\to\infty}\left|\frac{\partial^2}{(\partial x)^2}\int a^{\tau,m}\left[(w^{\tau,m}x+b)_{\tau}^{m}\right]\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}-\int a^{m}(w^{m})^{2}\rho_{m}^{*}(a,w,-w^{m}x)\mathrm{d}a\mathrm{d}w\right|\leq\frac{C}{m\lambda},$$

where C is a constant independent of (m,τ,β,λ).

Proof of Lemma 4.5.1. First, we show that

$$\begin{aligned}\lim_{\tau\to\infty}\Bigg|\int a^{\tau,m}&\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}^{m}\right]\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\&-\int a^{m}(w^{m})^{2}\rho_{m}^{*}(a,w,-w^{m}x)\mathrm{d}a\mathrm{d}w\Bigg|\leq\frac{C}{m\lambda}.\end{aligned} \tag{4.22}$$

Recall the definition of the activation $(\cdot)_{m}^{\tau}$ provided in (4.2). We can decompose the integral into two pieces with respect to the domain of truncation and obtain

$$\begin{aligned}&\int a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}^{m}\right]\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\&=\int_{w^{\tau,m}x+b\leq x_m}a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}\right]\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\&+\int_{w^{\tau,m}x+b>x_m}a^{\tau,m}(w^{\tau,m})^{2}\left[\frac{\partial^2}{(\partial u)^2}\phi_{\tau,m}(u)\Big|_{u=w^{\tau,m}x+b}\right]\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}.\end{aligned} \tag{4.23}$$

Let us focus on the first term in the RHS of (4.23). The second derivative has the following form

$$\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}=(w^{\tau,m})^{2}\cdot\frac{\tau e^{\tau(w^{\tau,m}x+b)}}{\left(e^{\tau(w^{\tau,m}x+b)}+1\right)^{2}}>0.$$

Thus, the following chain of equalities holds

$$\begin{aligned}
&\int_{w^{\tau,m}x+b\leq x_m} a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_\tau\right]\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\
=&\int_{w^{\tau,m}x+b\leq x_m} a^{\tau,m}(w^{\tau,m})^2\cdot\frac{\tau e^{\tau(w^{\tau,m}x+b)}}{\left(e^{\tau(w^{\tau,m}x+b)}+1\right)^2}\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\
=&\int \mathbb{1}_{\{y\leq\tau x_m\}}\cdot a^{\tau,m}(w^{\tau,m})^2\cdot\frac{e^y}{(e^y+1)^2}\rho^*_{\tau,m}\left(a,w,\frac{y}{\tau}-w^{\tau,m}x\right)\mathrm{d}(a,w,y),
\end{aligned}$$

where in the last step we have performed the change of variables $y=\tau(w^{\tau,m}x+b)$. By Lemma B.1.4, we have that, as $\tau\to\infty$, $\rho^*_{\tau,m}(\boldsymbol{\theta})$ converges to $\rho^*_m(\boldsymbol{\theta})$ pointwise in $\boldsymbol{\theta}$. Furthermore, as $\tau\to\infty$, $a^{\tau,m}$ converges to a^m for any a, and $w^{\tau,m}$ converges to w^m for any w. Thus, as the Gibbs distributions $\rho^*_{\tau,m}(\boldsymbol{\theta})$ and $\rho^*_m(\boldsymbol{\theta})$ are continuous with respect to $\boldsymbol{\theta}$, we have that

$$\begin{aligned}
&\lim_{\tau\to\infty}\left[\mathbb{1}_{\{y\leq\tau x_m\}}\cdot a^{\tau,m}(w^{\tau,m})^2\cdot\frac{e^y}{(e^y+1)^2}\rho^*_{\tau,m}\left(a,w,\frac{y}{\tau}-w^{\tau,m}x\right)\right]\\
&=a^m(w^m)^2\cdot\frac{e^y}{(e^y+1)^2}\rho^*_m\left(a,w,-w^m x\right).
\end{aligned}$$

Furthermore, combining (B.4) and (B.5) from Lemma B.1.2, we get the following bound

$$\rho^*_{\tau,m}(\boldsymbol{\theta})\leq C'\exp\left(-\frac{\beta\lambda\|\boldsymbol{\theta}\|_2^2}{2}\right), \tag{4.24}$$

for some constant $C'>0$ independent of $\boldsymbol{\theta}$ and τ. Thus, we have

$$\begin{aligned}
&|a^{\tau,m}|(w^{\tau,m})^2\cdot\frac{e^y}{(e^y+1)^2}\rho^*_{\tau,m}\left(a,w,\frac{y}{\tau}-w^{\tau,m}x\right)\\
\leq& C'm^3\cdot\frac{e^y}{(e^y+1)^2}\cdot\exp\left(-\frac{\beta\lambda(a^2+w^2)}{2}\right),
\end{aligned}$$

which is integrable in (y,a,w). Hence, by using the Dominated Convergence theorem and integrating out y using Tonelli's theorem, we have

$$\begin{aligned}
\lim_{\tau\to\infty}\Bigg|&\int_{w^{\tau,m}x+b\leq x_m} a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_\tau\right]\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\
&-\int a^m(w^m)^2\rho^*_m(a,w,-w^m x)\mathrm{d}a\mathrm{d}w\Bigg|=0.
\end{aligned}\tag{4.25}$$

Now, by triangle inequality, it remains to show that the absolute value of the second term in the RHS of (4.23) can be upper bounded by $\mathcal{O}\left(\frac{1}{m\lambda}\right)$ as $\tau\to\infty$. Recall that, by construction,

$$|\phi''_{\tau,m}(x)|\leq\frac{1}{m^2},\quad |a^{\tau,m}|\leq m,\quad |w^{\tau,m}|\leq|w|,$$

for any $x>x_m$ and any $(a,w)\in\mathbb{R}^2$. Thus, the following upper bound holds

$$\begin{aligned}
&\lim_{\tau\to\infty}\int_{w^{\tau,m}x+b>x_m}|a^{\tau,m}|(w^{\tau,m})^2\left|\frac{\partial^2}{(\partial u)^2}\phi_{\tau,m}(u)\Big|_{u=w^{\tau,m}x+b}\right|\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\
&\leq\frac{1}{m}\lim_{\tau\to\infty}\int w^2\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}.
\end{aligned}\tag{4.26}$$

In addition, we have the following pointwise convergence of the integrand

$$\lim_{\tau\to\infty} w^2\rho^*_{\tau,m}(\boldsymbol{\theta}) = w^2\rho^*_m(\boldsymbol{\theta}).$$

Furthermore, by using (4.24), we conclude that the integrand can be dominated by an integrable function. Hence, an application of the Dominated Convergence theorem gives that

$$\frac{1}{m}\lim_{\tau\to\infty}\int w^2\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} = \frac{1}{m}\int w^2\rho^*_m(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \le \frac{C''}{m\lambda}, \tag{4.27}$$

where the last inequality follows from Lemma B.1.2, which gives that $M(\rho^*_m) < C''/\lambda$ for some $C'' > 0$ that is independent of (m, λ). By combining (4.23), (4.25), (4.26) and (4.27), we conclude that (4.22) holds. Finally, by using a standard line of arguments, i.e., Mean Value theorem and Dominated Convergence, the derivative can be pushed inside the integral sign, which finishes the proof. □

Next, we study the set on which (4.21) might grow unbounded. In particular, in Lemma 4.5.3, we provide an upper bound on the measure of the set $\overline{\Omega}_j(m, \beta, \lambda)$ defined in (4.10)-(4.11). To do so, we will first show that the partition function of ρ^*_m is uniformly bounded in m, as stated and proved below.

Lemma 4.5.2 (Uniform bound on partition function)**.** *Consider $\sigma^*(\boldsymbol{\theta}, x) = a^{\tau,m}(w^{\tau,m}x + b)^m_\tau$ or $\sigma^*(\boldsymbol{\theta}, x) = a^m(w^m x + b)^m_+$, and let $\rho^*_{\sigma^*}$ be the Gibbs distribution with activation σ^*. Then, the following upper bound holds for its partition function $Z_{\sigma^*}(\beta, \lambda)$:*

$$\ln Z_{\sigma^*}(\beta,\lambda) \le \beta C + 1 + 3\log\frac{8\pi}{\beta\lambda},$$

where $C > 0$ is a constant independent of $(m, \tau, \beta, \lambda)$.

Proof of Lemma 4.5.2. Let $R^{\sigma^*}_i(\rho^*_{\sigma^*})$ be defined as follows

$$R^{\sigma^*}_i(\rho^*_{\sigma^*}) := -\frac{1}{M}\left(y_i - y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i)\right).$$

By substituting the form (2.38) of the Gibbs distribution into the free energy functional (2.36), we have that

$$\begin{aligned}\mathcal{F}^{\sigma^*}(\rho) = \frac{1}{2M}\sum_{i=1}^{M}\left(y_i - y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i)\right)^2 + \frac{\lambda}{2}M(\rho^*_{\sigma^*})\\ -\int\sum_{i=1}^{M}\left[R^{\sigma^*}_i(\rho^*_{\sigma^*})\cdot\sigma^*(x_i,\boldsymbol{\theta})\right]\rho^*_{\sigma^*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} - \frac{\lambda}{2}\int\|\boldsymbol{\theta}\|_2^2\rho^*_{\sigma^*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} - \frac{1}{\beta}\ln Z_{\sigma^*}(\beta,\lambda).\end{aligned}$$

Note that, by Fubini's theorem, we can interchange summation and integration in the first integral, since the activation and the labels are bounded. By using also the definition of

$R_i^{\sigma^*}(\rho^*_{\sigma^*})$, we have that

$$\begin{aligned}
\mathcal{F}^{\sigma^*}(\rho) &= \frac{1}{2M}\sum_{i=1}^{M} y_i^2 + \frac{1}{2M}\sum_{i=1}^{M}\left(y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i)\right)^2 - \frac{1}{M}\sum_{i=1}^{M} y_i \cdot y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i) + \frac{\lambda}{2}M(\rho^*_{\sigma^*}) \\
&\quad - \frac{1}{M}\sum_{i=1}^{M}\left(y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i)\right)^2 + \frac{1}{M}\sum_{i=1}^{M} y_i \cdot y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i) - \frac{\lambda}{2}M(\rho^*_{\sigma^*}) - \frac{1}{\beta}\ln Z_{\sigma^*}(\beta,\lambda) \\
&= -\frac{1}{\beta}\ln Z_{\sigma^*}(\beta,\lambda) - \frac{1}{2M}\sum_{i=1}^{M}\left(y^{\sigma^*}_{\rho^*_{\sigma^*}}(x_i)\right)^2 + \frac{1}{2M}\sum_{i=1}^{M} y_i^2 \\
&\leq -\frac{1}{\beta}\ln Z_{\sigma^*}(\beta,\lambda) + \frac{1}{2M}\sum_{i=1}^{M} y_i^2 \\
&\leq -\frac{1}{\beta}\ln Z_{\sigma^*}(\beta,\lambda) + C,
\end{aligned}$$

where $C > 0$ is independent of $(m, \tau, \beta, \lambda)$. From Lemma 10.2 in [MMN18], we obtain that, for any $\rho \in \mathcal{K}$,

$$\mathcal{F}(\rho) \geq \frac{1}{2}R(\rho) + \frac{\lambda}{4}M(\rho) - \frac{1}{\beta}\left[1 + 3\log\frac{8\pi}{\beta\lambda}\right] \geq -\frac{1}{\beta}\left[1 + 3\log\frac{8\pi}{\beta\lambda}\right],$$

where the last inequality follows from non-negativity of $R(\rho)$ and $M(\rho)$. Combining the upper and lower bounds gives

$$-\frac{1}{\beta}\ln Z_{\sigma^*}(\beta,\lambda) + C \geq -\frac{1}{\beta}\left[1 + 3\log\frac{8\pi}{\beta\lambda}\right].$$

After a rearrangement, we have

$$\ln Z_{\sigma^*}(\beta,\lambda) \leq \beta C + 1 + 3\log\frac{8\pi}{\beta\lambda},$$

which concludes the proof. □

In order to bound the measure of $\overline{\Omega}_j(m,\beta,\lambda)$, the idea is to combine the upper bound on the partition function of Lemma 4.5.2 with a lower bound that diverges in m unless $|\overline{\Omega}_j(m,\beta,\lambda)|$ vanishes. In particular, we derive a lower bound with the structure of a Gaussian integral which grows unbounded for a certain set of inputs. This set of inputs corresponds to the scenario when the Gaussian covariance has non-positive eigenvalues, and it can be expressed as the set in which the polynomials f_j and f^j defined in (4.8) are non-negative. For brevity, we suppress the dependence of Ω_j and Ω^j on (m,β,λ) in the proofs below.

Lemma 4.5.3 (Bound on measure of cluster set)**.** *Assume that condition **B1** holds. For $j \in \{0,\dots,M\}$, let Ω^j and Ω_j be defined as in (4.11). Then,*

$$|\Omega^j|,\ |\Omega_j| \leq K_1\frac{e^{\beta K_2}}{m^2}, \tag{4.28}$$

where $K_1, K_2 > 0$ is independent of (m,β,λ).

Proof of Lemma 4.5.3. We start with the proof for Ω^j. For $j = M$, the corresponding polynomial $f^M(x)$ is equal to $1 + x^2$ and therefore $|\Omega^M| = 0$. Let us now consider the case $j \neq M$, and assume that $\mu(\Omega^j) > 0$. (If that's not the case, the claim trivially holds.)

Note that, as $f^j(x)$ is a polynomial of degree at most two in x, Ω^j is the union of at most two intervals. Then, the following set has a non-zero Lebesgue measure in $\mathbb{R}^2$:

$$\Omega := \{(w,b) \in \mathbb{R}_+ \times \mathbb{R} : b = -w^m x,\ 0 < w < m,\ x \in \Omega^j\}.$$

Now, we can lower bound the partition function as

$$\begin{aligned} Z_m(\beta,\lambda) &\geq \int_{\{|a|<m\}\times\Omega} \exp\left\{-\frac{\beta\lambda}{2}\left[\frac{2}{\lambda}\sum_{i=1}^{M} R_i^m(\rho_m^*)\cdot a^m(w^m x_i + b)_+^m + \|\boldsymbol{\theta}\|_2^2\right]\right\} \mathrm{d}\boldsymbol{\theta} \\ &= \int_{\{|a|<m\}\times\Omega} \exp\left\{-\frac{\beta\lambda}{2}\left[\frac{2}{\lambda}\sum_{i=j+1}^{M} R_i^m(\rho_m^*)\cdot a^m(w^m x_i + b) + \|\boldsymbol{\theta}\|_2^2\right]\right\} \mathrm{d}\boldsymbol{\theta}. \end{aligned} \tag{4.29}$$

Here, the equality in the second line follows from the following observation: if $i \in [j]$ and $(w,b) \in \Omega$, then $w^m x_i + b \leq 0$ and therefore $(w^m x_i + b)_+^m = 0$; if $i > j$ and $(w,b) \in \Omega$, then $0 < w^m x_i + b < m^2$ ($|x|, |x_i| \leq L$, hence $|x_i - x| \leq m$, as L is a numerical constant independent of m and m is sufficiently large by assumption **B1**) and therefore $(w^m x_i + b)_+^m = w^m x_i + b$ for all $(w,b) \in \Omega$. Thus, after the change of variables $(a,w,b) \mapsto (a,w,-w^m x)$ and an application of Tonelli's theorem, the RHS in (4.29) reduces to

$$\int_{x\in\Omega^j}\int_{\{|a|<m\}\times\{0<w<m\}} w\cdot\exp\left\{-\frac{\beta\lambda}{2}\left[2aw(B^j - A^j x) + a^2 + w^2(1+x^2)\right]\right\}\mathrm{d}(a,w)\mathrm{d}x. \tag{4.30}$$

Here the coefficients A^j and B^j are defined as per (4.9). The term under the exponent can be rewritten as

$$2aw(B^j - A^j x) + a^2 + w^2(1+x^2) = \begin{bmatrix} a & w \end{bmatrix} \Sigma^{-1} \begin{bmatrix} a \\ w \end{bmatrix},$$

with

$$\Sigma^{-1} = \begin{bmatrix} 1 & (B^j - A^j x) \\ (B^j - A^j x) & 1 + x^2 \end{bmatrix}.$$

By definition of Ω^j in conjunction with Sylvester's criterion, we have that Σ^{-1} has a non-positive eigenvalue with corresponding eigenvector

$$\lambda_- = \frac{1}{2}\left(-\sqrt{4(B^j - A^j x)^2 + x^4} + x^2 + 2\right) \leq 0, \quad v_- = \left(-\frac{x^2 + \sqrt{4(B^j - A^j x)^2 + x^4}}{2(B^j - A^j x)}, 1\right).$$

Furthermore, the other eigenvalue with corresponding eigenvector is given by

$$\lambda_+ = \frac{1}{2}\left(\sqrt{4(B^j - A^j x)^2 + x^4} + x^2 + 2\right) > 0, \quad v_+ = \left(-\frac{x^2 - \sqrt{4(B^j - A^j x)^2 + x^4}}{2(B^j - A^j x)}, 1\right).$$

Note that v_- and v_+ are orthogonal, and consider the following change of variables for the integral

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} v_-/\|v_-\|_2 \\ v_+/\|v_+\|_2 \end{bmatrix}\begin{bmatrix} a \\ w \end{bmatrix} = Q^T \begin{bmatrix} a \\ w \end{bmatrix} \Leftrightarrow Q\mathbf{z} = \begin{bmatrix} a \\ w \end{bmatrix} \Leftrightarrow \begin{bmatrix} a(\mathbf{z}) \\ w(\mathbf{z}) \end{bmatrix} := Q\mathbf{z}.$$

As the matrix Q is unitary, the quantity in (4.30) can be rewritten as

$$\int_{x\in\Omega^j}\int_{\{|a(\mathbf{z})|<m\}\times\{0<w(\mathbf{z})<m\}} w(\mathbf{z})\cdot\exp\left\{-\frac{\beta\lambda}{2}\left[\lambda_- z_1^2 + \lambda_+ z_2^2\right]\right\}\mathrm{d}\mathbf{z}\mathrm{d}x,$$

as the determinant of the Jacobian is 1 for any unitary linear transformation. As $\lambda_- \le 0$, this quantity is lower bounded by

$$\int_{x\in\Omega^j}\int_{\{|a(\mathbf{z})|<m\}\times\{0<w(\mathbf{z})<m\}} w(\mathbf{z})\cdot\exp\left\{-\frac{\beta\lambda}{2}\left[\lambda_+ z_2^2\right]\right\}\mathrm{d}\mathbf{z}\mathrm{d}x. \tag{4.31}$$

Notice that $\|v_-\| \ge 1$, $\|v_+\| \ge 1$ and $w(\mathbf{z}) = z_1/\|v_-\|_2 + z_2/\|v_+\|_2$. Thus, picking $z_1 \in (0, m/2]$ and $z_2 \in (0, m/2]$ ensures that $0 < w(\mathbf{z}) < m$. Furthermore, these conditions on $\mathbf{z}$ do not violate the requirement on $a(\mathbf{z})$, since $|a(\mathbf{z})| \le |z_1| + |z_2| \le m$. Consequently, as the integrand is non-negative, the integral in (4.31) is lower bounded by

$$\int_{x\in\Omega^j}\int_{\{0<z_1<m/2\}\times\{0<z_2<m/2\}} w(\mathbf{z})\cdot\exp\left\{-\frac{\beta\lambda}{2}\left[\lambda_+ z_2^2\right]\right\}\mathrm{d}\mathbf{z}\mathrm{d}x. \tag{4.32}$$

By Lemma B.1.5, $|R_i^m(\rho_m^*)|$ is bounded by a constant independent of (m, β, λ), since $\lambda < C_3$ from condition **B1**. Hence, $\lambda|A^j x - B^j|$ is also uniformly bounded in (m, β, λ). This, in particular, implies that

$$\lambda\cdot\lambda_+ \le K_1,$$

where $K_1 > 0$ is independent of (m, β, λ). Furthermore, by definition of Ω^j, $|B^j - A^j x| > 1$, which implies that $\|v_+\|_2$ and $\|v_-\|_2$ are also upper bounded by a constant $K_2 > 0$ independent of (m, β, λ), and therefore

$$w(z) \le \frac{z_1 + z_2}{K_2}.$$

With this in mind, we can then further lower bound the integral in (4.32) by

$$\begin{aligned}
&\int_{x\in\Omega^j}\int_{\{0<z_1<m/2\}\times\{0<z_2<m/2\}} \frac{1}{K_2}(z_1+z_2)\cdot\exp\left\{-\frac{K_1\beta}{2}\cdot z_2^2\right\}\mathrm{d}\mathbf{z}\mathrm{d}x\\
&= |\Omega^j|\int_{\{0<z_1<m/2\}\times\{0<z_2<m/2\}} \frac{1}{K_2}(z_1+z_2)\cdot\exp\left\{-\frac{K_1\beta}{2}\cdot z_2^2\right\}\mathrm{d}\mathbf{z}\\
&\ge |\Omega^j|\int_{\{0<z_1<m/2\}\times\{0<z_2<m/2\}} \frac{1}{K_2}z_1\cdot\exp\left\{-\frac{K_1\beta}{2}\cdot z_2^2\right\}\mathrm{d}\mathbf{z}\\
&= \frac{|\Omega^j|}{K_2}\left[\frac{m^2}{8}\sqrt{\frac{\pi}{2K_1\beta}}\mathrm{erf}\left(\frac{m\sqrt{K_1\beta}}{2\sqrt{2}}\right)\right]\\
&\ge |\Omega^j|\frac{K_3\, m^2}{\sqrt{\beta}},
\end{aligned} \tag{4.33}$$

where $K_3 > 0$ is independent of (m, β, λ) and in the last passage we have used that $\mathrm{erf}\left(\frac{m\sqrt{K_1\beta}}{2\sqrt{2}}\right) \ge 1/10$ for sufficiently large m and β. By combining (4.33) with the upper bound on the partition function given by Lemma 4.5.2, the desired result immediately follows and the proof for Ω^j is complete.

In regards to the argument for Ω_j, for $j = 0$ the result trivially holds, since $f_0(x) = 1 + x^2$ and, thus, $|\Omega_0| = 0$. For $j > 0$, the partition function can be lower bounded by

$$\int_{\{|a|<m\}\times\Omega}\exp\left\{-\frac{\beta\lambda}{2}\left[\frac{2}{\lambda}\sum_{i=1}^{j}R_i^m(\rho_m^*)\cdot a^m(w^m x_i + b) + \|\boldsymbol{\theta}\|_2^2\right]\right\}\mathrm{d}\boldsymbol{\theta}, \tag{4.34}$$

where the set Ω is defined on non-positive w and $x \in \Omega_j$, i.e.,

$$\Omega := \{(w, b) \in \mathbb{R}_+ \times \mathbb{R} : b = -w^m x,\ -m < w < 0,\ x \in \Omega_j\}.$$

The rest of the argument remains the same by noting that with the change of variable

$$(a, w, b) \mapsto (-a, -w, w^m x)$$

the quantity in (4.34) is equal to

$$\int_{x\in\Omega_j} \int_{\{|a|<m\}\times\{0<w<m\}} w \cdot \exp\left\{-\frac{\beta\lambda}{2}\left[2aw(B_j - A_j x) + a^2 + w^2(1+x^2)\right]\right\} \mathrm{d}(a,w)\mathrm{d}x,$$

which is exactly as in (4.30), but with $x \in \Omega_j$ and the polynomial $(B_j - A_j x)$ in place of $x \in \Omega^j$ and the polynomial $(B^j - A^j x)$. □

In order to control the magnitude of (4.21), it is also necessary to understand the behavior of the polynomials defined in (4.8). The worst case scenario, in terms of presenting a challenge to bounding the curvature, corresponds to f^j or f_j being arbitrarily close to zero on the whole area outside of cluster set. In fact, this would imply that the Gaussian-like integral arising in the computation of (4.21) has arbitrary small eigenvalues. More specifically, our plan is to exploit the following bound for $x \in I_j \setminus \overline{\Omega}_j(m, \beta, \lambda)$:

$$\begin{aligned}|(4.21)| \leq C \int |a|w^2 \Bigg[&\exp\left\{-\frac{\beta\lambda}{2} \cdot f^j(x) \cdot (a^2 + w^2)\right\} \\ &+ \exp\left\{-\frac{\beta\lambda}{2} \cdot f_j(x) \cdot (a^2 + w^2)\right\}\Bigg] \mathrm{d}\theta.\end{aligned} \tag{4.35}$$

Now, the RHS of (4.35) diverges (and, therefore, the bound is useless), if either of the polynomials is arbitrarily close to zero outside of the cluster set. Fortunately, we are able to prove that this cannot happen: in Lemma 4.5.5 we show that $f^j(x)$ and $f_j(x)$ can be small only when x approaches the cluster set, i.e.,

$$f^j(x), f_j(x) \geq \min\{C^j(x), C_j(x), 1\},$$

where $C^j(x), C_j(x)$ are defined in (4.13) and, because of the condition on their coefficients $\{K_i\}_{i=1}^4$, they cannot be arbitrarily close to 0 in any interval I_j.

As a preliminary step towards the proof of Lemma 4.5.5, we show an auxiliary result for polynomials of a certain form. Fix some interval $I = [I_l, I_r] \subset \mathbb{R}$. Given two quantities $a, b \in \mathbb{R}$, consider the following polynomial of degree at most two

$$P_2(x) := (1 - a^2) \cdot x^2 + 2ab \cdot x + (1 - b^2), \quad x \in I, \tag{4.36}$$

where we suppress the dependence on (a, b), i.e., $P_2(x; a, b) = P_2(x)$, for more compact notation. In addition, let Ω_+ be the subset of I on which P_2 is strictly positive, i.e.,

$$\Omega_+ := \{x \in I : P_2(x) > 0\}.$$

For a fixed small constant $C_\Omega > 0$, define the set of admissible coefficients as follows

$$\mathcal{U} := \{(a, b) \in \mathbb{R}^2 : |\Omega_+| \geq C_\Omega\}. \tag{4.37}$$

Given $(a, b) \in \mathcal{U}$ and $x \in \Omega_+$, we define the critical point x_c of the polynomial P_2 associated with x and Ω_+ in the same fashion as in Definition 4.4.1, after replacing $f^j(\cdot)$ with $P_2(\cdot)$ and $I_j \setminus \Omega^j(m, \beta, \lambda)$ with Ω_+. Notice that, since Ω_+ has strictly positive Lebesgue measure for $(a, b) \in \mathcal{U}$, the critical point is well-defined and, in particular, $x_c \in I$ always holds.

Lemma 4.5.4 (Lower bound on polynomial). *Fix some C_Ω such that $\mathcal{U}$, as defined in (4.37), is of positive measure. Pick some interval $(a, b) \in \mathcal{U}$. Let $x \in \Omega_+$ and x_c be the critical point associated to x. Then, the following holds*

$$P_2(x) \geq \alpha_2(x - x_c)^2 + \alpha_1|x - x_c| + \alpha_0, \tag{4.38}$$

where $\alpha_0, \alpha_1, \alpha_2 \geq 0$ and at least one of them is lower bounded by a strictly positive constant depending on C_Ω but independent of the choice of $(a, b) \in \mathcal{U}$.

We defer the proof of Lemma 4.5.4 to Appendix B.1.3. Recall the definition of the polynomial $f^j(x)$ given in (4.8), and notice that expression can be rearranged such that $f^j(x)$ is in the form of (4.36), namely

$$f^j(x) = 1 + x^2 - (A^j x - B^j)^2 = (1 - (A^j)^2)x^2 + 2A^j B^j x + (1 - (B^j)^2).$$

In this view, the following result follows from Lemma 4.5.4.

Lemma 4.5.5 (Well-defined quadratic form). *Assume that $(A^j, B^j) \in \mathcal{U}$, i.e., $|I_j \setminus \Omega^j|$ is lower bounded by a positive constant. Given $x \in I_j \setminus \Omega^j$, let x_c be the critical point associated to x. Then, we have that*

$$f^j(x) \geq C^j(x) := \gamma_1(x - x_c)^2 + \gamma_2, \tag{4.39}$$

where $\gamma_1, \gamma_2 > 0$ and either $\gamma_1 > \varepsilon$ or $\gamma_2 > \varepsilon$ for some $\varepsilon > 0$ that is independent of (A^j, B^j) but depending on C_Ω as appearing in the definition of $\mathcal{U}$.

Proof of Lemma 4.5.5. Note that $I_j \setminus \Omega^j$ is the set in which f^j is strictly positive. Hence, since $|I_j \setminus \Omega^j|$ is lower bounded by a positive constant independent of A^j, B^j, we can apply Lemma 4.5.4 to get

$$f^j(x) \geq \alpha_2(x - x_c)^2 + \alpha_1|x - x_c| + \alpha_0,$$

where $\alpha_0, \alpha_1, \alpha_2 \geq 0$ and at least one of them is lower bounded by a strictly positive constant independent of (A^j, B^j). Thus, since each term of the RHS above is non-negative, we get

$$f^j(x) \geq \alpha_i|x - x_c|^i + \alpha_0,$$

where $i = \arg\max_{j \in \{1,2\}} \alpha_j$. Furthermore, as $|x - x_c| \leq |I_j|$, we have

$$f^j(x) \geq \frac{\alpha_i}{|I_j|^{2-i}}|x - x_c|^2 + \alpha_0.$$

Now, either α_i or α_0 as well as $1/|I_j|$ are lower bounded by strictly positive constants independent of (A^j, B^j). Thus, taking $\gamma_1 = \alpha_i/|I_j|^{2-i}$ and $\gamma_2 = \alpha_0$ concludes the proof. □

Let us point out that, although ε does not depend on the values of $(A^j, B^j) \in \mathcal{U}$, the position of a critical point x_c *depends* on (A^j, B^j).

In a similar fashion, we define $\bar{\mathcal{U}}$ to be the set of admissible (A_j, B_j) as in (4.37), and given $x \in I_j \setminus \Omega_j$, we let $\bar{x}_c$ be the critical point associated to x and Ω_j. Then, a result analogous to Lemma 4.5.5 holds for $f_j(x)$:

$$f_j(x) \geq C_j(x) := \gamma_3(x - \bar{x}_c)^2 + \gamma_4, \tag{4.40}$$

where $\gamma_3, \gamma_4 > 0$ and either $\gamma_3 > \varepsilon$ or $\gamma_4 > \varepsilon$ for some $\varepsilon > 0$ that is independent of the choice of $(A_j, B_j) \in \bar{\mathcal{U}}$.

The last ingredient for the proof of the vanishing curvature phenomenon is the control of the decay of the partition function $Z_m(\beta, \lambda)$ as $\beta \to 0$.

Lemma 4.5.6 (Lower bound on partition function independent of m)**.** *Assume that condition **B1** holds. Then,*

$$Z_m(\beta, \lambda) \geq \frac{C}{\sqrt{\beta^3 \lambda^{3/2}}},$$

for some $C > 0$ that is independent of (m, β, λ).

The proof of Lemma 4.5.6 is deferred to Appendix B.1.2. At this point, we are ready to provide an upper bound on the magnitude of (4.21).

Lemma 4.5.7 (Integral upper bound)**.** *Assume that condition **B1** holds. Furthermore, assume that $m > e^{\beta K_2}$, where K_2 is given in (4.28). Fix $j \in \{0, \dots, M\}$. Then, for any $x \in I_j \setminus (\Omega^j \cup \Omega_j)$,*

$$\left|\int a^m (w^m)^2 \rho_m^*(a, w, -w^m x) \mathrm{d}a\mathrm{d}w\right| \leq \frac{K}{\beta \lambda^{7/4} (\bar{C}^j(x))^2},$$

where $K > 0$ is independent of (m, β, λ), $\bar{C}^j(x) := \min\{C_j(x), C^j(x), 1\}$, and $C^j(x), C_j(x)$ are given by (4.39) and (4.40), respectively.

Proof of Lemma 4.5.7. Note that the following upper bound holds

$$\left|\int a^m (w^m)^2 \rho_m^*(a, w, -w^m x) \mathrm{d}a\mathrm{d}w\right| \leq I(x) := \int |a^m| (w^m)^2 \rho_m^*(a, w, -w^m x) \mathrm{d}a\mathrm{d}w.$$

Let us now decompose the integral $I(x)$ depending on the sign of w, i.e.,

$$Z_m(\beta, \lambda) \cdot I(x) = I^j(x) + I_j(x),$$

where

$$I^j(x) := \int_{\{a \in \mathbb{R}\} \times \{w \geq 0\}} |a^m| (w^m)^2 \exp\left\{-\beta \Psi^j(a, w, \rho_m^*)\right\} \mathrm{d}a\mathrm{d}w,$$
$$I_j(x) := \int_{\{a \in \mathbb{R}\} \times \{w < 0\}} |a^m| (w^m)^2 \exp\left\{-\beta \Psi_j(a, w, \rho_m^*)\right\} \mathrm{d}a\mathrm{d}w,$$

and, recalling the form of $\rho_m^*(a, w, -w^m x)$ from (4.5), the corresponding potentials are given by

$$\Psi^j(a, w, \rho) = \sum_{i=j+1}^{M} R_i^m(\rho) \cdot a^m w^m (x_i - x) + \frac{\lambda}{2}\left\{a^2 + w^2 + (w^m)^2 x^2\right\},$$
$$\Psi_j(a, w, \rho) = \sum_{i=1}^{j} R_i^m(\rho) \cdot a^m w^m (x_i - x) + \frac{\lambda}{2}\left\{a^2 + w^2 + (w^m)^2 x^2\right\}.$$

By recalling from (4.9) the definitions of A^j, A_j, B^j and B_j, we obtain the following upper bounds.

$$I^j(x) \leq 2\int_{\{a \geq 0\} \times \{w \geq 0\}} a w^2 \exp\left\{-\frac{\beta\lambda}{2}\left[-2a w^m |B^j - A^j x| + a^2 + w^2 + (w^m)^2 x^2\right]\right\} \mathrm{d}a\mathrm{d}w, \tag{4.41}$$

$$I_j(x) \leq 2\int_{\{a \geq 0\} \times \{w < 0\}} a w^2 \exp\left\{-\frac{\beta\lambda}{2}\left[2a w^m |B_j - A_j x| + a^2 + w^2 + (w^m)^2 x^2\right]\right\} \mathrm{d}a\mathrm{d}w. \tag{4.42}$$

Let us analyze the RHS of (4.41). This term can be rewritten as

$$\begin{aligned}2\int_{\{a\geq 0\}\times\{w\geq 0\}} & aw^2\exp\left\{-\frac{\beta\lambda}{2}\left[-2aw^m|A^jx-B^j|+a^2+(w^m)^2(A^jx-B^j)^2\right]\right\}\\ &\cdot\exp\left\{-\frac{\beta\lambda}{2}\left[w^2+(w^m)^2x^2-(w^m)^2(A^jx-B^j)^2\right]\right\}\mathrm{d}a\mathrm{d}w.\end{aligned} \tag{4.43}$$

Note that

$$|\Omega^j|\leq\frac{K_1\,e^{\beta K_2}}{m^2}\leq\frac{K_1}{e^{\beta K_2}},$$

where the first inequality follows from Lemma 4.5.3, and the second inequality uses that $m>e^{\beta K_2}$. Therefore, for sufficiently large β, $|\Omega^j|$ is smaller than $|I_j|/2$, and therefore $|I_j\setminus\Omega^j|$ is lower bounded by $|I_j|/2$. At this point, we can apply Lemma 4.5.5 which gives that $1+x^2-(A^jx-B^j)^2\geq C^j(x)\geq\bar{C}^j(x):=\min\{C^j(x),C_j(x),1\}$. Thus, (4.43) is upper bounded by

$$\begin{aligned}2\int_{\{a\geq 0\}\times\{w\geq 0\}} & aw^2\exp\left\{-\frac{\beta\lambda}{2}\left(a-|B^j-A^jx|w^m\right)^2\right\}\\ &\cdot\exp\left\{-\frac{\beta\lambda}{2}\left[w^2-(w^m)^2(1-\bar{C}^j(x))\right]\right\}\mathrm{d}a\mathrm{d}w\\ =2\int_{\{w\geq 0\}} & w^2\exp\left\{-\frac{\beta\lambda}{2}\left[w^2-(w^m)^2(1-\bar{C}^j(x))\right]\right\}\sqrt{\frac{2\pi}{\beta\lambda}}\mathbb{E}\left[(A)_+\right]\mathrm{d}w,\end{aligned} \tag{4.44}$$

where $A\sim\mathcal{N}(|B^j-A^jx|w^m,(\beta\lambda)^{-1})$. Furthermore, the following chain of inequalities hold:

$$\mathbb{E}\left[(A)_+\right]\leq\mathbb{E}\left[|A|\right]\leq\sqrt{\mathbb{E}\left[A^2\right]}=\sqrt{|B^j-A^jx|^2(w^m)^2+\frac{1}{\beta\lambda}}, \tag{4.45}$$

where the second passage follows from Jensen's inequality. By using (4.45), the RHS of (4.44) is upper bounded by

$$\begin{aligned}\frac{2\sqrt{2\pi}}{\sqrt{\beta\lambda}}\int_{\{w\geq 0\}} & \sqrt{(B^j-A^jx)^2(w^m)^2+\frac{1}{\beta\lambda}}\\ &\cdot w^2\exp\left\{-\frac{\beta\lambda}{2}\left[w^2-(w^m)^2(1-\bar{C}^j(x))\right]\right\}\mathrm{d}w.\end{aligned}$$

Applying Lemma 4.5.5 again to obtain $(A^jx-B^j)^2\leq 1+x^2-\bar{C}^j(x)\leq 1+x^2$ and noting by definition that $(w^m)^2\leq w^2$, we now upper bound this last term by

$$\begin{aligned}&2\sqrt{2\pi}\int_{\{w\geq 0\}}\sqrt{\frac{w^2(1+x^2)}{\beta\lambda}+\frac{1}{\beta^2\lambda^2}}\;\cdot\;w^2\exp\left\{-\frac{\beta\lambda}{2}\left[w^2-(w^m)^2(1-\bar{C}^j(x))\right]\right\}\mathrm{d}w\\ &\leq 2\sqrt{2\pi}\int_{\{w\in\mathbb{R}\}}\sqrt{\frac{w^2(1+x^2)}{\beta\lambda}+\frac{1}{\beta^2\lambda^2}}\;\cdot w^2\exp\left\{-\frac{\beta\lambda}{2}\left[\bar{C}^j(x)\cdot w^2\right]\right\}\mathrm{d}w\\ &\leq 2\sqrt{2\pi}\int_{\{w\in\mathbb{R}\}}\left(\sqrt{\frac{w^2(1+x^2)}{\beta\lambda}}+\sqrt{\frac{1}{\beta^2\lambda^2}}\right)\;\cdot w^2\exp\left\{-\frac{\beta\lambda}{2}\left[\bar{C}^j(x)\cdot w^2\right]\right\}\mathrm{d}w,\end{aligned} \tag{4.46}$$

where in the second line we use that $1-\bar{C}^j(x)\geq 0$ and again that $(w^m)^2\leq w^2$, and in the third line we use that $\sqrt{u+v}\leq\sqrt{u}+\sqrt{v}$.

Finally, computing explicitly the last integral gives the following upper bound on the RHS of (4.41) and consequently on $I^j(x)$:

$$I^j(x) \leq 4\pi\sqrt{\frac{1}{\beta^2\lambda^2}} \cdot \sqrt{\frac{1}{(\bar{C}^j(x))^3\beta^3\lambda^3}} + 2\sqrt{2\pi}\sqrt{\frac{1+x^2}{\beta\lambda}}\sqrt{\frac{1}{(\bar{C}^j(x))^4\beta^4\lambda^4}}.$$

By following the similar passages, we obtain the same upper bound for $I_j(x)$. By using the lower bound on the partition function shown in Lemma 4.5.6, we conclude that

$$I(x) = \frac{I^j(x) + I_j(x)}{Z_m(\beta, \lambda)} \leq \frac{K}{\beta\lambda^{7/4}(\bar{C}^j(x))^2},$$

where $K > 0$ is independent of (m, β, λ), and the proof is complete. □

The proof of Theorem 3 is an immediate consequence of the results presented so far.

Proof of Theorem 3. The proof of (4.14) follows from Lemmas 4.5.1 and 4.5.7, and the proof of (4.16) follows from Lemma 4.5.3. □

4.5.3 Proof of Theorem 4

To summarize, at this point we have shown that as $\beta \to \infty$ the second derivative of the predictor vanishes outside the cluster set, and that the size of the cluster set shrinks to concentrate on at most 3 points per prediction interval. With these results in mind, we are ready to provide the proof for Theorem 4.

Proof of Theorem 4. The predictor evaluated at the Gibbs distribution is given by

$$y_n(x) = \int a^{\tau,m}(w^{\tau,m}x + b)_\tau^m \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta},$$

where $n = (\tau, m, \beta, \lambda)$ denotes the aggregated index and we suppress the dependence on (β, λ) in $\rho^*_{\tau,m}$ for convenience. By Lemma B.1.6, there exists $\tau(m, \beta, \lambda)$ such that, for any $\tau > \tau(m, \beta, \lambda)$,

$$M(\rho^*_{\tau,m}) \leq C, \tag{4.47}$$

for some $C > 0$ independent of $(\tau, m, \beta, \lambda)$. We start by showing that the family of predictors $\{y_n\}$ is equi-Lipschitz for $\infty > \tau > \tau(m, \beta, \lambda)$. First, note that

$$\frac{\partial}{\partial x}y_n(x) = \int \frac{\partial}{\partial x}\Big[a^{\tau,m}(w^{\tau,m}x + b)_\tau^m\Big]\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}, \tag{4.48}$$

since the derivative can be pushed inside by the same line of arguments as given in the proof of Lemma 4.5.1. Next, we have that, by construction of the activation, the following holds

$$\int \frac{\partial}{\partial x}\Big[a^{\tau,m}(w^{\tau,m}x + b)_\tau^m\Big]\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \leq C_1\int |a^{\tau,m}w^{\tau,m}|\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta},$$

where, from here on, $C_1 > 0$ denotes a generic constant which might change from line to line, but is independent of $(\tau, m, \beta, \lambda)$. By construction, for any $u \in \mathbb{R}$, it holds that $|u^{\tau,m}| \leq |u|$. Thus, we have that

$$\int \frac{\partial}{\partial x}\Big[a^{\tau,m}(w^{\tau,m}x + b)_\tau^m\Big]\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \leq C_1\int |aw|\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}.$$

Using the Cauchy-Schwartz inequality and (4.47), we obtain that

$$\int \frac{\partial}{\partial x}\Big[a^{\tau,m}(w^{\tau,m}x+b)_{\tau}^{m}\Big]\rho_{\tau,m}^{*}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \le C_1 M(\rho_{\tau,m}^{*}) \le C_1. \tag{4.49}$$

By combining (4.48) and (4.49), we have shown that the family $\{y_n\}$ for $\tau > \tau(m,\beta,\lambda)$ is equi-Lipschitz, as the derivatives are uniformly bounded. By using a similar argument, we can show that the same result holds for the predictor itself, i.e., for all $x \in \bigcup_{j=0}^{M} I_j$, $y_n(x)$ is uniformly bounded.

Note that Theorem 3 considers the curvature of points outside the cluster set, and it gives an upper bound which diverges when $\bar{C}^j(x)$ approaches 0 for some $j \in [M]$. Thus, our next step is to develop the analytical machinery to make this scenario impossible. Let us recall Definitions (4.13) and (4.15). Then, by Lemma 4.5.5, we have that

$$\bar{C}^j(x) \ge \min\{\gamma_1(x-x_c)^2+\gamma_2,\ \gamma_3(x-\bar{x}_c)^2+\gamma_4\},$$

where $\gamma_1,\gamma_2,\gamma_3,\gamma_4 > 0$ and $\min\{\max\{\gamma_1,\gamma_2\},\max\{\gamma_3,\gamma_4\}\} > \varepsilon$, for some $\varepsilon > 0$ that is independent of (m,β,λ). Let us focus on the term $\gamma_1(x-x_c)^2+\gamma_2$. If $\gamma_2 = 0$ or it approaches 0 (as $m,\beta \to \infty$), then we extend $\Omega^j(m,\beta,\lambda)$ as

$$\mathrm{ext}_\delta(\Omega^j(m,\beta,\lambda)) := \left\{x \in I_j : \min_{x' \in \Omega^j(m,\beta,\lambda)\cup\{x_c\}} |x-x'| < \delta\right\}.$$

Note that adding the singleton $\{x_c\}$ to the argument of the $\min$ allows us to also cover the case in which $\Omega^j(m,\beta,\lambda)$ is empty. Otherwise, i.e., if $\gamma_2 > \varepsilon$ for some $\varepsilon > 0$ that is independent of (m,β,λ), the upper bound on the curvature does not diverge and we set $\mathrm{ext}_\delta(\Omega^j(m,\beta,\lambda)) := \Omega^j(m,\beta,\lambda)$. In a similar fashion, we define the extension of $\Omega_j(m,\beta,\lambda)$ by $\mathrm{ext}_\delta(\Omega_j(m,\beta,\lambda))$.

Let $\bar{\Omega}^j_{\text{ext}}$ be the union of $\mathrm{ext}_\delta(\Omega^j(m,\beta,\lambda))$ and $\mathrm{ext}_\delta(\Omega_j(m,\beta,\lambda))$, where we drop the explicit dependence of $\bar{\Omega}^j_{\text{ext}}$ on (δ,m,β,λ) for convenience. Then, since f^j and f_j are polynomials of degree two, the extended set $\bar{\Omega}^j_{\text{ext}}$ (just like $\bar{\Omega}^j$) is the union of at most three disjoint open intervals, i.e.,

$$\bar{\Omega}^j_{\text{ext}} = A_1^j \cup A_2^j \cup A_3^j,$$

where $\{A_i^j\}_{i=1}^3$ denote such (possibly empty) open intervals. Furthermore, $I_j \setminus \bar{\Omega}^j_{\text{ext}}$ is the union of at most three disjoint closed intervals, i.e.,

$$I_j \setminus \bar{\Omega}^j_{\text{ext}} = B_1^j \cup B_2^j \cup B_3^j,$$

where $\{B_i^j\}_{i=1}^3$ denote such (possibly empty) closed intervals.

At this point, we are ready to show that, for all closed intervals $\{B_i^j\}_{i=1}^3$, the predictor y_n can be approximated arbitrarily well by a linear function (which may be different in different closed intervals). Note that y_n is twice continuously differentiable for $\tau < \infty$, and fix $\tilde{x} \in B_i^j$. Then, by combining Taylor's theorem with the result of Theorem 3, we obtain that, for any $x \in B_i^j$,

$$\lim_{\tau\to\infty} |y_n(x) - y_n(\tilde{x}) - y_n'(\tilde{x})(x-\tilde{x})| \le \mathcal{O}\left(\frac{1}{m\lambda} + \frac{1}{\delta^4 \cdot \beta\lambda^{7/4}}\right), \tag{4.50}$$

where we use that $|x-x_c| \ge \delta$ by construction of the extended set $\bar{\Omega}^j_{\text{ext}}$. Let us define

$$f_n^i(x) = y_n(\tilde{x}) - y_n'(\tilde{x})(x-\tilde{x}).$$

Then, by picking a sufficiently small δ, (4.50) implies that, as $m\lambda \to \infty$ and $\beta\lambda^{7/4} \to \infty$, for all $x \in B_i^j$,

$$|y_n(x) - f_n^i(x)| \to 0. \tag{4.51}$$

We remark that, as shown previously, the coefficients $y_n(\tilde{x})$ and $y_n'(\tilde{x})$ are uniformly bounded in absolute value.

Let us now consider the open intervals $\{A_i^j\}_{i=1}^3$. For any $x \in A_i^j$, let

$$x' = \underset{y \notin A_i^j}{\arg\min} |x - y|,$$

and note that, by definition, $x' \in B_{\tilde{\imath}}^j$ for some $\tilde{\imath} \in \{1, 2, 3\}$. By picking the linear approximation $f_n^{\tilde{\imath}}$ that corresponds to $B_{\tilde{\imath}}^j$ and by using the triangle inequality, we obtain that

$$\begin{aligned} |y_n(x) - f_n^{\tilde{\imath}}(x)| &\le |y_n(x) - y_n(x')| + |y_n(x') - f_n^{\tilde{\imath}}(x')| + |f_n^{\tilde{\imath}}(x') - f_n^{\tilde{\imath}}(x)| \\ &\le \mathcal{O}\left(|x - x'| + |y_n(x') - f_n^{\tilde{\imath}}(x')|\right), \end{aligned} \tag{4.52}$$

where the second inequality is due to the fact that the families $\{y_n\}$ and $\{f_n^i\}$ are equi-Lipschitz. From (4.51) the second term in the RHS in (4.52) vanishes. As for the first term, by construction of the extension, together with the result of Lemma 4.5.3, we have that

$$|x - x'| \le \mathcal{O}\left(\frac{e^{\beta K_2}}{m^2} + \delta\right),$$

for some $K_2 > 0$ independent of (m, β, λ). Thus, by picking a sufficiently small δ and $m > e^{\beta K_2}$, we conclude that the first term in the RHS in (4.52) also vanishes.

So far, we have showed that, both inside and outside of the extension of the cluster set, the predictor y_n is well approximated by linear functions. It remains to prove that the linear pieces connect, i.e., there exists $\hat{x} \in \bar{\Omega}_{\text{ext}}^j$ such that, for two neighboring linearities f_n^i and f_n^{i+1} (possibly belonging to different intervals), the following holds

$$f_n^i(\hat{x}) - f_n^{i+1}(\hat{x}) = 0.$$

This claim follows from Lipschitz arguments similar to those presented above, and the proof is complete. □

4.5.4 Proof of Corollary 4.4.3

At this point, we have proved a result about the structure of the predictor coming from the minimizer of the free energy (2.36). By using the mean-field analysis in [MMN18], we finally show that this structural result holds for the predictor obtained from a wide two-layer ReLU network.

Proof of Corollary 4.4.3. First, we show that, as $t \to \infty$, the second derivative of the predictor evaluated on the solution ρ_t of the flow (2.33) converges to the same quantity evaluated on the Gibbs minimizer $\rho_{\tau,m}^*$. To do so, we decompose the integral involving ρ_t as in Lemma

4.5.1 (cf. (4.23)):

$$\begin{aligned}
&\int a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}^{m}\right]\rho_t(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\
&\quad=\int_{w^{\tau,m}x+b\le x_m} a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}\right]\rho_t(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\\
&\qquad+\int_{w^{\tau,m}x+b>x_m} a^{\tau,m}(w^{\tau,m})^2\left[\frac{\partial^2}{(\partial u)^2}\phi_{\tau,m}(u)\bigg|_{u=w^{\tau,m}x+b}\right]\rho_t(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}. \qquad (4.53)
\end{aligned}$$

Next, we show that a technical condition bounding the free energy at initialization appearing in the statement of Theorem 4 in [MMN18] is satisfied under the assumption $M(\rho_0)<\infty$ and $H(\rho_0)>-\infty$. Recalling the sandwich bound for the truncated soft-plus activation (4.3) and the fact that that $\tau\ge 1$ by condition **B1**, an application of Cauchy-Schwarz inequality gives

$$R^{\tau,m}(\rho_0)<CM(\rho_0)+C'<\infty,$$

where $C,C'>0$ are some numerical constants independent of (τ,m). This readily implies that

$$\mathcal{F}^{\tau,m}(\rho_0)<\infty,$$

since λ and β^{-1} are upper-bounded by assumption **B1**.

Now we can apply Theorem 4 in [MMN18] to conclude that, as $t\to\infty$,

$$\rho_t\rightharpoonup\rho^*_{\tau,m}.$$

Thus, as the terms inside the integrals in (4.53) are all bounded for fixed (τ,m,β,λ), by definition of weak convergence, we get that, as $t\to\infty$,

$$\int a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}^{m}\right]\rho_t(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\to\int a^{\tau,m}\left[\frac{\partial^2}{(\partial x)^2}(w^{\tau,m}x+b)_{\tau}^{m}\right]\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}.$$

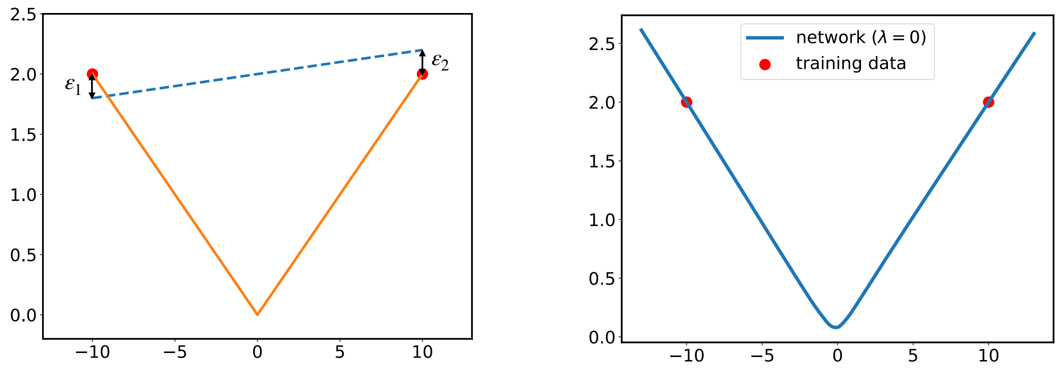
Consequently, since the derivative operator can be pushed inside by the same arguments as in Lemma 4.5.1, we have that, as $t\to\infty$, the following pointwise convergence holds

$$\frac{\partial^2}{(\partial x)^2}\int a^{\tau,m}(w^{\tau,m}x+b)_{\tau}^{m}\rho_t(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\to\frac{\partial^2}{(\partial x)^2}\int a^{\tau,m}(w^{\tau,m}x+b)_{\tau}^{m}\rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}. \qquad (4.54)$$

Next, we show that the second derivative of the predictor obtained from the two-layer ReLU network also converges to the same limit. Recall that $\sigma^*(x,\boldsymbol{\theta})=a^{\tau,m}(w^{\tau,m}x+b)_{\tau}^{m}$. Then, by Theorem 3 in [MMN18], we have that, almost surely, as $N\to\infty,\ \varepsilon_N\to 0$

$$\frac{\partial^2}{(\partial x)^2}\left[\frac{1}{N}\sum_{i=1}^{N}\sigma^*\left(x,\boldsymbol{\theta}_i^{\lfloor t/\varepsilon\rfloor}\right)\right]\to\frac{\partial^2}{(\partial x)^2}\int a^{\tau,m}(w^{\tau,m}x+b)_{\tau}^{m}\rho_t(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \qquad (4.55)$$

along any sequence $\{\varepsilon_N\}$ such that $\varepsilon_N\log(N/\varepsilon_N)\to 0$ and $N/\log(N/\varepsilon_N)\to\infty$. By combining (4.54) and (4.55), we obtain that the desired convergence result holds for the LHS of (4.54).



(a) (b)

Figure 4.6: (a) The orange curve represents the function $f^*(x)$ which interpolates the training data (red dots) and exhibits a knot at the point $(0, 0)$; the blue dashed curve is linear in the interval between the training points ($\varepsilon_1 = -0.2$, $\varepsilon_2 = 0.2$). (b) We run noiseless SGD ($\beta = \infty$) with no regularization ($\lambda = 0$) for a two-layer ReLU network with $N = 500$ neurons, trained on the dataset (4.57). The resulting estimator (in blue) approaches the piecewise linear function $f^*(x)$ with a knot between the two training data points.

Another application of Theorem 3 of [MMN18], together with the fact that the second moment of the flow solution ρ_t is uniformly bounded along the sequence $t \to \infty$ (cf. Lemma 10.2 in [MMN18], following Proposition 4.1 in [JKO98]), gives that the gradients

$$\frac{\partial}{\partial x}\left[\frac{1}{N}\sum_{i=1}^{N}\sigma^*\left(x, \boldsymbol{\theta}_i^{\lfloor t/\varepsilon \rfloor}\right)\right]$$

are almost surely uniformly bounded. This fact, in turn, implies that the corresponding predictor is almost surely equi-Lipschitz. In a similar fashion, we also have that the predictor itself is almost surely uniformly bounded in absolute value.

At this point, the desired result follows from the same line of arguments as in the proof of Theorem 4. □

4.6 Knots Inside the Interval

In this section, we provide an explicit example of a 2-point dataset such that the SGD solution exhibits a change of tangent (or "knot") *inside* the training interval. To do so, we will show that neural networks implementing a linear function without knots on the prediction interval *cannot* minimize the free energy (2.36). To simplify the analysis, throughout the section we omit the limits in (τ, m), i.e., we consider directly ReLU activations (this corresponds to taking $\tau = m = \infty$). Similar arguments apply to the case of sufficiently large parameters τ and m.

4.6.1 Noiseless Regime

We start with the case of noiseless SGD training, i.e., $\beta = +\infty$. Here, the free energy has no entropy penalty and it can be expressed as

$$\mathcal{F}_\infty(\rho) = \frac{1}{2}R(\rho) + \frac{\lambda}{2}M(\rho). \tag{4.56}$$

We consider the following dataset which consists of two points:

$$\mathcal{D} = \{(-\bar{x}, \bar{y}), (\bar{x}, \bar{y})\} = \{(-10, 2), (10, 2)\}. \tag{4.57}$$

Let $f^*(x)$ be the piecewise linear function that interpolates the training data $\{(-\bar{x}, \bar{y}), (\bar{x}, \bar{y})\}$ and passes through the point $(0, 0)$, where it exhibits a knot (see the orange curve in Figure 4.6a). Note that

$$f^*(x) = \int a(wx + b)_+ \rho^*(a, w, b) \mathrm{d}a\mathrm{d}w\mathrm{d}b,$$

where

$$\rho^*(a, b, w) = \frac{1}{2}\left[\delta_{\left(\sqrt{2\frac{\bar{y}}{\bar{x}}}, -\sqrt{2\frac{\bar{y}}{\bar{x}}}, 0\right)}(a, w, b) + \delta_{\left(\sqrt{2\frac{\bar{y}}{\bar{x}}}, \sqrt{2\frac{\bar{y}}{\bar{x}}}, 0\right)}(a, w, b)\right], \tag{4.58}$$

and $\delta_{(a_0, w_0, b_0)}$ denotes the Dirac delta function centered at (a_0, w_0, b_0). Note that $R(\rho^*) = 0$ and $M(\rho^*) = \frac{2}{5}$. Thus, the free energy is given by

$$\mathcal{F}_\infty(\rho^*) = \frac{1}{2}R(\rho^*) + \frac{\lambda}{2}M(\rho^*) = \frac{\lambda}{5}. \tag{4.59}$$

Let $f(x)$ be a linear function on the interval $[-\bar{x}, \bar{x}]$ such that $f(-\bar{x}) = \bar{y} + \varepsilon_1$ and $f(\bar{x}) = \bar{y} + \varepsilon_2$ (see the blue dashed line in Figure 4.6a), and let ρ be the corresponding distribution of the parameters, i.e.,

$$f(x) = \int a(wx + b)_+ \rho(a, w, b) \mathrm{d}a\mathrm{d}w\mathrm{d}b. \tag{4.60}$$

In the rest of this section, we will show that, for all $\lambda \leq 1$,

$$\min_{\varepsilon_1, \varepsilon_2} \mathcal{F}_\infty(\rho) > \mathcal{F}_\infty(\rho^*). \tag{4.61}$$

In words, the minimizer of the free energy cannot be a linear function on the interval $[-\bar{x}, \bar{x}]$. As f is linear, we have that

$$f(x) = \frac{\varepsilon_2 - \varepsilon_1}{2\bar{x}}(x - \bar{x}) + \bar{y} + \varepsilon_2,$$

which implies that

$$f(0) = \bar{y} + \frac{\varepsilon_1 + \varepsilon_2}{2} = \int a(b)_+ \rho(a, b) \mathrm{d}a\mathrm{d}b. \tag{4.62}$$

First, we consider the case $f(0) = 0$. From (4.62), we have that $\varepsilon_1 + \varepsilon_2 = -2\bar{y}$. Hence,

$$\mathcal{F}_\infty(\rho) \geq \frac{1}{2}R(\rho) = \frac{1}{4}(\varepsilon_1^2 + \varepsilon_2^2) \geq \frac{1}{8}(\varepsilon_1 + \varepsilon_2)^2 = \frac{\bar{y}^2}{2} = 2. \tag{4.63}$$

By combining (4.63) and (4.59), we conclude that (4.61) holds for all $\lambda \leq 1$ (under the additional restriction $f(0) = 0$).

Next, we consider the case $f(0) \neq 0$. By using (4.62) and applying Cauchy-Schwarz inequality, we have that

$$|f(0)| = \left|\int a(b)_+ \rho(a, b) \mathrm{d}a\mathrm{d}b\right| = |\mathbb{E}[a(b)_+]| \leq \sqrt{\mathbb{E}[a^2]\,\mathbb{E}[(b)_+^2]} \quad \Longrightarrow \quad \mathbb{E}[a^2] \geq \frac{(f(0))^2}{\mathbb{E}[(b)_+^2]}.$$

With this in mind, we can lower bound the regularization term as

$$M(\rho) \geq \mathbb{E}[a^2] + \mathbb{E}[b^2] \geq \frac{(f(0))^2}{\mathbb{E}[(b)_+^2]} + \mathbb{E}[b^2] \geq \frac{(f(0))^2}{\mathbb{E}[(b)_+^2]} + \mathbb{E}[(b)_+^2] \geq 2|f(0)| = 2\left|\bar{y} + \frac{\varepsilon_1 + \varepsilon_2}{2}\right|,$$

where the last inequality follows from the fact that $g(t) = (f(0))^2/t + t$ is minimized over $t \geq 0$ by taking $t = |f(0)|$. Therefore, we have that

$$\mathcal{F}_\infty(\rho) \geq \frac{1}{4}(\varepsilon_1^2 + \varepsilon_2^2) + \lambda\left|\bar{y} + \frac{\varepsilon_1 + \varepsilon_2}{2}\right|.$$

Note that, for a fixed value of the sum $\varepsilon_1 + \varepsilon_2$, the quantity $\varepsilon_1^2 + \varepsilon_2^2$ is minimized when $\varepsilon_1 = \varepsilon_2$. Thus, by recalling that $\bar{y} = 2$, we have

$$\mathcal{F}_\infty(\rho) \geq \min_{\varepsilon}\left\{\frac{1}{2}\varepsilon^2 + \lambda|2 + \varepsilon|\right\}. \tag{4.64}$$

One can readily verify that, for any $\lambda \leq 2$, the minimizer is given by $\varepsilon^* = -\lambda$. Thus,

$$\mathcal{F}_\infty(\rho) \geq 2\lambda - \frac{\lambda^2}{2} \geq \frac{3\lambda}{2} > \frac{\lambda}{5} = \mathcal{F}_\infty(\rho^*), \tag{4.65}$$

where the first inequality uses (4.64) and that the minimizer is $\varepsilon^* = -\lambda$, and the next two inequalities use that $\lambda \geq 1$. Merging two cases regarding $f(0)$, we conclude that (4.61) holds, as desired.

4.6.2 Low Temperature Regime

We now focus on the case of noisy SGD with temperature β^{-1}. Here, the free energy can be expressed as

$$\mathcal{F}_\beta(\rho) = \frac{1}{2}R(\rho) + \frac{\lambda}{2}M(\rho) - \beta^{-1}H(\rho). \tag{4.66}$$

We consider the two-point dataset (4.57) and we recall that $f^*(x)$ has a knot inside the training interval. In this section we will show that the following two results hold for all $\lambda \leq 1$:

(i) There exists a sequence of distributions $\{\rho_\beta^*\}_\beta$ such that, for any $x \in [-\bar{x}, \bar{x}]$,

$$\lim_{\beta\to\infty} \int a(wx + b)_+ \rho_\beta^*(a, w, b) \mathrm{d}a\mathrm{d}w\mathrm{d}b = f^*(x), \tag{4.67}$$

and

$$\limsup_{\beta\to\infty} \mathcal{F}_\beta(\rho_\beta^*) \leq \frac{\lambda}{5}. \tag{4.68}$$

(ii) Let ρ be a distribution such that the function $f(x)$ given by (4.60) is linear in the interval $[-\bar{x}, \bar{x}]$. Pick a sequence of distributions $\{\rho_\beta\}_\beta$ such that $\rho_\beta \rightharpoonup \rho$ and for any $x \in [-\bar{x}, \bar{x}]$,

$$\lim_{\beta\to\infty} \int a(wx + b)_+ \rho_\beta(a, w, b) \mathrm{d}a\mathrm{d}w\mathrm{d}b = f(x). \tag{4.69}$$

Then, we have that

$$\liminf_{\beta\to\infty} \mathcal{F}_\beta(\rho_\beta) > \frac{\lambda}{5}. \tag{4.70}$$

Combining these two results gives that, for sufficiently large β, the minimizer of the free energy (4.66) cannot yield a linear estimator on the interval between the two data points. In Figure 4.6b, we represent the function obtained by training via SGD a two-layer ReLU network with 500 neurons on the dataset (4.57). Clearly, the blue curve approaches the piecewise linear function $f^*(x)$, which contains a knot inside the interval $[-10, 10]$. The plot represented in the Figure corresponds to the case with no regularization ($\lambda = 0$), but similar results are obtained for small (but non-zero) regularization.

Proof of *(i)*. Let ρ^*_β be defined as

$$\rho^*_\beta = \frac{1}{2}\left[\mathcal{N}\left(\left[\sqrt{2\frac{\bar{y}}{\bar{x}}}, -\sqrt{2\frac{\bar{y}}{\bar{x}}}, 0\right], \beta^{-1} I_{3\times 3}\right) + \mathcal{N}\left(\left[\sqrt{2\frac{\bar{y}}{\bar{x}}}, \sqrt{2\frac{\bar{y}}{\bar{x}}}, 0\right], \beta^{-1} I_{3\times 3}\right)\right],$$

where $\mathcal{N}(\mu, \Sigma)$ denotes the multivariate Gaussian distribution with mean μ and covariance Σ. As $\beta \to \infty$, we have that $\rho^*_\beta \rightharpoonup \rho^*$, where ρ^* is given by (4.58). However, weak convergence does not suffice for pointwise convergence of the corresponding estimators, since the function $\sigma^*(x) = a(wx + b)_+$ is unbounded (in x). To solve this issue, we observe that the fourth moment of ρ^*_β is uniformly bounded as $\beta \to \infty$. Thus, by the de la Vallée Poussin criterion (see e.g. [HR11]), we have that the sequence of random variables $\{\|X_\beta\|_2^2\}_\beta$ is uniformly integrable, with $X_\beta \sim \rho^*_\beta$. Consider a ball $B_r = \{\mathbf{v} \in \mathbb{R}^3 : \|\mathbf{v}\|_2 \le r\}$, for $r > \sqrt{4\bar{y}/\bar{x}}$. Then, we have

$$\begin{aligned}\left|\int_{\mathbb{R}^3} a(wx+b)_+(\rho^*_\beta(a,w,b) - \rho^*(a,w,b))\mathrm{d}a\mathrm{d}w\mathrm{d}b\right| \\ \le \left|\int_{B_r} a(wx+b)_+(\rho^*_\beta(a,w,b) - \rho^*(a,w,b))\mathrm{d}a\mathrm{d}w\mathrm{d}b\right| \\ + \left|\int_{\mathbb{R}^3\setminus B_r} a(wx+b)_+\rho^*_\beta(a,w,b)\mathrm{d}a\mathrm{d}w\mathrm{d}b\right|,\end{aligned} \tag{4.71}$$

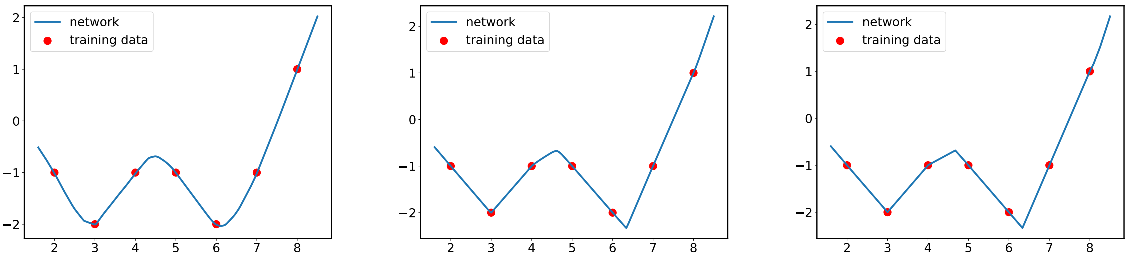
where we have used that the support of ρ^* lies inside the ball B_r. The first term in the RHS of (4.71) vanishes as $\beta \to \infty$ by weak convergence, since the function $a(wx + b)_+$ is bounded inside B_r. For the second term, we have that, for any $x \in [-\bar{x}, \bar{x}]$,

$$\begin{aligned}\left|\int_{\mathbb{R}^3\setminus B_r} a(wx+b)_+\rho^*_\beta(a,w,b)\mathrm{d}a\mathrm{d}w\mathrm{d}b\right| &\le \int_{\mathbb{R}^3\setminus B_r} (|aw|\cdot|x| + |ab|)\rho^*_\beta(a,w,b)\mathrm{d}a\mathrm{d}w\mathrm{d}b \\ &\le C\int_{\mathbb{R}^3\setminus B_r} (a^2+b^2+w^2)\rho^*_\beta(a,w,b)\mathrm{d}a\mathrm{d}w\mathrm{d}b,\end{aligned}$$

where $C > 0$ is a constant independent of (β, r). Since the sequence $\{\|X_\beta\|_2^2\}_\beta$ is uniformly integrable, we can make the RHS arbitrary small by picking a sufficiently large r (uniformly for all β). As a result, (4.67) readily follows. Note that (4.67) immediately implies that, as $\beta \to \infty$, $R(\rho^*_\beta) \to R(\rho^*) = 0$. Furthermore, with similar arguments we obtain that, as $\beta \to \infty$, $M(\rho^*_\beta) \to M(\rho^*)$. By convexity of the differential entropy, we have that $H(\frac{1}{2}\rho_1 + \frac{1}{2}\rho_2) \ge \frac{1}{2}H(\rho_1) + \frac{1}{2}H(\rho_2)$. Hence, $H(\rho^*_\beta) \ge C\log(2\pi e/\beta)$, where $C > 0$ is independent of β. By combining these bounds on $R(\rho^*_\beta)$, $M(\rho^*_\beta)$ and $H(\rho^*_\beta)$, we conclude that

$$\limsup_{\beta\to\infty} \mathcal{F}_\beta(\rho^*_\beta) \le \mathcal{F}_\infty(\rho^*),$$

which, combined with (4.59), completes the proof of (4.68). □



(a) $\beta^{-1} = 0.005$ (b) $\beta^{-1} = 0.0001$ (c) $\beta^{-1} = 0$

Figure 4.7: Functions learnt by a two-layer ReLU network with $N = 500$ neurons, for different values of the temperature parameter β^{-1}. The regularization coefficient λ is set to zero.

Proof of *(ii)*. From (4.69), we obtain that $\lim_{\beta\to\infty} R(\rho_\beta) = R(\rho)$. As the second moment is lower-semicontinuous and bounded from below, we have that $\liminf_{\beta\to\infty} M(\rho_\beta) \geq M(\rho)$. Furthermore, Lemma 10.2 in [MMN18] implies that

$$\mathcal{F}_\beta(\rho_\beta) \geq \frac{1}{2}R(\rho_\beta) + \frac{\lambda}{4}M(\rho_\beta) - \beta^{-1}(1 + 3\log 8\pi) + \beta^{-1}\log(\beta\lambda).$$

By combining these bounds, we have that

$$\liminf_{\beta\to\infty} \mathcal{F}_\beta(\rho_\beta) \geq \frac{1}{2}R(\rho) + \frac{\lambda}{4}M(\rho). \tag{4.72}$$

By replicating the argument leading to (4.65) (but now with regularization coefficient $\lambda/2$ instead of λ), we obtain that the RHS of (4.72) can be lower bounded as

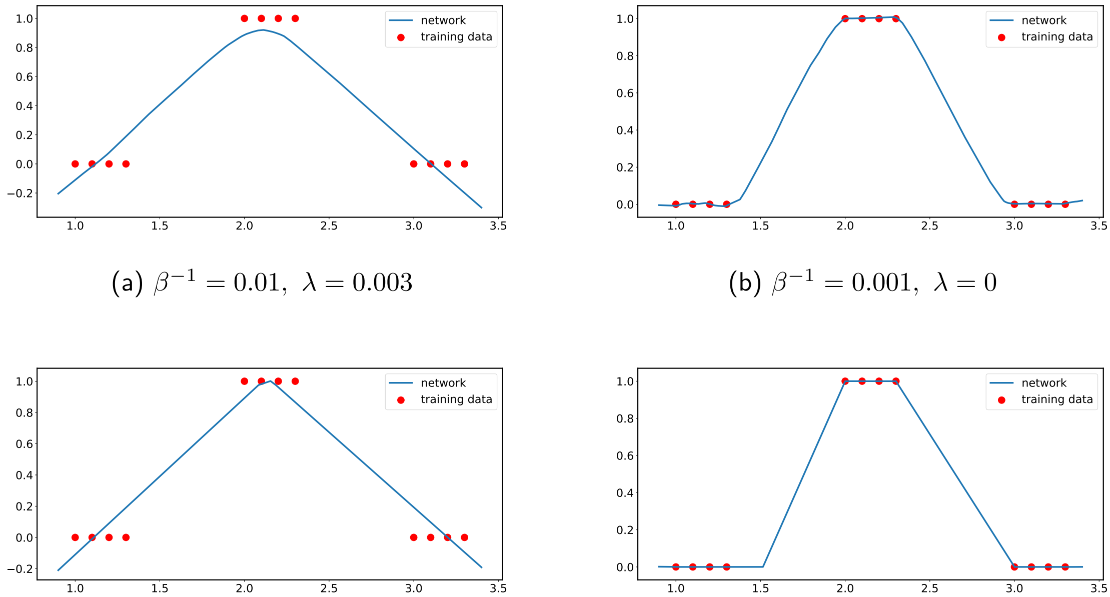
$$\frac{1}{2}R(\rho) + \frac{\lambda}{4}M(\rho) \geq \lambda - \frac{\lambda^2}{8} \geq \frac{7\lambda}{8} > \frac{\lambda}{5}, \tag{4.73}$$

for all $\lambda \leq 1$. Then, the desired result follows from (4.72) and (4.73). □

4.7 Numerical Simulations

We consider training the two-layer neural network (2.25) with N neurons and ReLU activation functions, i.e., $\sigma^*(x, \boldsymbol{\theta}) = a(wx + b)_+$, with $\boldsymbol{\theta} = (a, w, b)$. We run the SGD iteration (2.30) (no momentum or weight decay, batch size equal to 1), and we plot the resulting predictor once the algorithm has converged. The results for two different unidimensional datasets are reported in Figures 4.7 and 4.8. In these experiments, we set $N = 500$ and we remark that the plots for wider networks ($N \in \{1000, 2000, 5000\}$) look identical. We also point out that the shape of the predictor does not change for different runs of the SGD algorithm (with different initializations, and order of the training samples). This is in agreement with the mean-field predictions when $\beta < \infty$, $\lambda > 0$ and the variance of the initialization does not depend on N. The same setup is employed to obtain the numerical results of Figure 4.1 and 4.6b, discussed in Section 4.1 and 4.6, respectively.

In Figure 4.7, we plot the shape of the function learnt by the network for different values of the temperature parameter β^{-1}. The learning rate is $s_k = 1$, the total number of training epochs required for SGD to converge is roughly 5×10^4, and no ℓ_2 regularization is enforced ($\lambda = 0$). As predicted by our theoretical findings, the predictor approaches a piecewise linear



(a) $\beta^{-1} = 0.01, \;\lambda = 0.003$

(b) $\beta^{-1} = 0.001, \;\lambda = 0$

(c) $\beta^{-1} = 0, \;\lambda = 0.003$

(d) $\beta^{-1} = 0, \;\lambda = 0$

Figure 4.8: Functions learnt by a two-layer ReLU network with $N = 500$ neurons, for different values of the temperature parameter β^{-1} and the regularization coefficient λ.

function whose number of tangent changes (or knots) is proportional to the number of training samples (and not to the width of the network): if $\beta^{-1} = 0.005$, the predictor is still rather smooth; if $\beta^{-1} = 10^{-4}$, the predictor sharpens, except for a smoother tangent change in the interval $[4, 5]$; and finally if $\beta = 0$, the predictor is piecewise linear. Let us highlight that the knots sometimes do not coincide with the training data points, as suggested by the results of Section 4.4 and demonstrated in the example of Section 4.6.

In Figure 4.8, we consider another dataset and plot the neural network predictor for four different pairs of (β^{-1}, λ). By comparing (a) with (b) and with the bottom plots (c)-(d), it is clear that the solution becomes increasingly piecewise linear as the noise decreases. Furthermore, the effect of regularization can be noticed by comparing plots (a)-(c) on the left with plots (b)-(d) on the right: adding an ℓ_2 penalty implies that the network does not fit the data and therefore the location of the knots changes.

4.8 Comparison with Related Work

The line of works [SESS19, EP21, OWSS20, PN20a] studies the properties of the minimizers of certain optimization objectives, and therefore these results are not directly connected to the dynamics of gradient descent algorithms. On the contrary, the goal of this chapter is to understand the implicit bias due to gradient descent, namely, to characterize the structure of the neural network predictor once the algorithm has converged. Another important difference lies in the fact that our ℓ_2 regularization involves all the parameters, including the bias b, while existing work does not regularize the biases of the network. This fact may lead to the qualitatively different behavior unveiled by our study. Going into detail, [EP21] show that the network that minimizes a regularized objective implements a linear spline. In contrast, our analysis suggests that the knots (i.e., abrupt changes in the tangent of the predictor) can occur at points different from the training samples. Let us also mention that [SESS19] and

[OWSS20] give an explicit form of the functional regularizer of the neural network solution, but it is not clear how to characterize the function class to which the solution belongs, e.g., whether the function implemented by the neural network is a cubic or linear spline. Furthermore, the upper bound on the number of knot points appearing in [PN20a] depends on the null space of a certain operator, and computing the dimension of this null space explicitly appears to be difficult.

The work by [WTS$^+$19] considers a noiseless setting with no regularization, and it studies the properties of gradient flow on the space of reduced parameters. In particular, the initial ReLU neurons depending on three parameters (a, b and w, in our notation) are mapped to a two-dimensional space, where each neuron is defined by its magnitude and angle. Then, it is proven that the Wasserstein gradient flow on this reduced space drives the activation points of the ReLU neurons to the training data. As a consequence, the solution found by SGD is piecewise linear and the knot points are located at a subset of the training samples. [BGVV20] consider SGD with label noise and no regularization, and show that, once the squared loss is close to zero, the algorithm minimizes an auxiliary quantity, i.e., the sum of the squared norms of the gradients evaluated at each training point. By instantiating this result in the case of a two-layer ReLU network with a skip connection, the authors show that the solution found by SGD is piecewise linear with the minimum amount of knots required to fit the data.

While our result shares some similarities with [WTS$^+$19] and [BGVV20], let us highlight some crucial differences. First, we note that [BGVV20] consider a two-layer network with a skip connection which fits the training data perfectly. In contrast, our two-layer model is standard (no skip connections) and the analysis does not require a perfect fit of the data, as we allow for non-vanishing ℓ_2 regularization. Furthermore, even when the regularization term is vanishing, our characterization does not lead to the minimum number of knots required to fit the data [as in BGVV20], and the knots are not necessarily located at the training points [as in WTS$^+$19]. In fact, our theoretical results suggest the presence of additional knot points, a feature that is confirmed in numerical simulations. The novel behavior that we unveil appears to be due to the differences in the setting and to the addition of (a possibly vanishing) ℓ_2 regularization term in the optimization. Concerning the proof techniques, the work by [BGVV20] exploits an Ornstein-Uhlenbeck like analysis, while this work tackles the increasingly popular mean-field regime. Our key technical contribution is to analyze the Gibbs minimizer of a certain free energy, while [WTS$^+$19] consider the gradient flow on reduced parameters and connect it to the flow on the full parameters via a specific type of initialization. Our analysis directly establishes a result for the full parameters, and it requires mild technical assumptions on the initialization. Finally, let us point out that it is an open problem to extend the approach of [WTS$^+$19] to a regularized objective, because of the non-injectivity of the mapping to the canonical parameters.

4.9 Concluding Remarks

We develop a new technique to characterize the implicit bias of gradient descent methods used to train overparameterized neural networks. In particular, we consider training a wide two-layer ReLU network via SGD for a univariate regression task and, by taking a mean field view, we show that the predictor obtained at convergence has a simple piecewise linear form. Our results hold in the regime of vanishingly small noise added to the SGD gradients, and handle both constant and vanishing ℓ_2 regularization. The analysis leads to an exact characterization of the number and location of the tangent changes (or knots) in the predictor: on each interval between consecutive training inputs, the number of knots is at most three. To obtain the desired result, we relate the distribution of the weights of the network once SGD has converged to the minimizer of a certain free energy. Then, we prove that the curvature of the predictor resulting from this minimizer vanishes everywhere except in a *cluster set*, which concentrates on at most three points per prediction interval. This novel strategy opens the way to several interesting directions. We discuss them below.

We focus on ReLU networks. However, only the following two properties of the activation appear to be crucial for the analysis: *(i)* its second derivative behaves like a Dirac delta, and *(ii)* its growth is at most linear. In fact, the first property reduces the computation of the curvature to an integral over a lower-dimensional subspace; and the second property leads to a uniform bound on the second moment of the network parameters. Hence, our approach may be extendable to a more general class of piecewise linear activations, although this would come at the cost of a more intricate structure for the cluster set containing the location of the tangent changes.

We focus on univariate regression. The natural ordering on one-dimensional features allows for a convenient characterization of the activation regions that correspond to each input conditioned on the sign of w. For larger input dimension, such a characterization appears to be cumbersome, as the structure of these regions is induced by the intersection of hyperplanes. Furthermore, in the setting considered in this work, the cluster set is the union of intervals where certain second-degree polynomials are non-positive. For multivariate regression, we expect the cluster set to be connected to the non-positive set of quadratic forms. Hence, the structure of the cluster set may be highly non-linear, and its concentration can occur on subspaces which are hard to define explicitly. Moreover, a naive estimate would suggest that in the higher-dimensional case the corresponding bound on the curvature of the predictor described in Theorem 3 should worst-case scale *exponentially* in the dimension of the inputs. This is suggested by the combinatorial complexity of the intersection of hyperplanes. However, it is important to note that the evidence [EP23] indicates that the exponential estimate is overly pessimistic. In particular, the corresponding bounds might actually behave *polynomially* in the *effective* dimension (rank) of the data.

We provide an upper bound on the number of tangent changes of the predictor. The numerical simulations of Section 4.6 suggest that one and two knots between consecutive training inputs can occur. Showing whether our theoretical bound of three knots is tight by providing an explicit example, or by proving a tighter bound of two, is an open question for possible future work. We also remark that, given the errors R_i of the neural network estimator at the data points, one can deduce the location of the knot points. Such implicit characterization is similar in spirit to the attractive/repulsive condition on the training points of [WTS$^+$19].

In conclusion, in this work we demonstrate how to exploit the Gibbs form of the minimizer in order to accurately characterize a functional property of the predictor learnt by the neural network using limiting arguments of the training process. The general spirit of this technique could potentially be informative in additional ways. For instance, utilizing the properties of the Gibbs distribution reached at convergence may be of additional interest for future study. We conjecture that this could yield insight into the stability of the predictor with respect to perturbations in the training data at *finite* temperature β.

CHAPTER 5

Fundamental Limits of Two-layer Autoencoders

Autoencoders are a popular model in many branches of machine learning and lossy data compression. However, their fundamental limits, the performance of gradient methods and the features learnt during optimization remain poorly understood, even in the two-layer setting. In fact, earlier work has considered either linear autoencoders or specific training regimes (leading to vanishing or diverging compression rates). This chapter addresses this gap by focusing on non-linear two-layer autoencoders trained in the challenging proportional regime in which the input dimension scales linearly with the size of the representation. Our results characterize the minimizers of the population risk, and show that such minimizers are achieved by gradient methods; their structure is also unveiled, thus leading to a concise description of the features obtained via training. For the special case of a sign activation function, our analysis establishes the fundamental limits for the lossy compression of Gaussian sources via (shallow) autoencoders. Finally, while the results are proved for Gaussian data, numerical simulations on standard datasets display the universality of the theoretical predictions.

5.1 Motivation and Outlook

Autoencoders represent a key building block in many branches of machine learning [KW14, RMW14], including generative modeling [BYAV13] and representation learning [TBL18]. Prompted by the fact that autoencoders learn succinct representations, neural autoencoding techniques have achieved remarkable success in lossy data compression, even outperforming classical methods, such as jpeg [BLS17, TSCH17, AMT^{+}17]. However, despite the large body of empirical work on neural autoencoders and compressors, basic theoretical questions remain poorly understood even in the shallow case:

> *What are the fundamental performance limits of autoencoders? Can we achieve such limits with gradient methods? What features does the optimization procedure learn?*

Prior work has focused either on linear autoencoders [BH89, KBGS19, GBLJ19], on the severely under-parameterized setting in which the input dimension is much larger than the number of neurons [RG22], or on specific training regimes (lazy training [NWH21] and mean-field regime

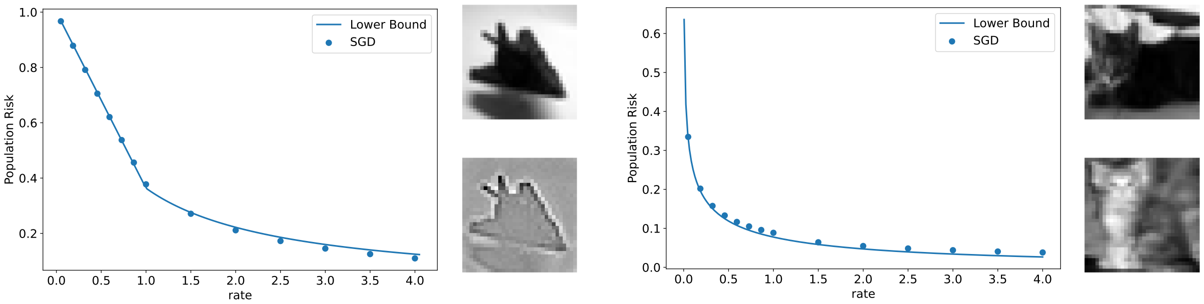


Figure 5.1: **Left plot.** Compression ($\sigma \equiv \mathrm{sign}$) of the grayscale CIFAR-10 "airplane" class with a two-layer autoencoder. The data is *whitened* so that $\boldsymbol{\Sigma} = \boldsymbol{I}$: on top, an example of a grayscale image; on the bottom, the corresponding whitening. The blue dots are the population risk obtained via SGD, and they agree well with the solid line corresponding to the lower bounds of Theorem 5 and Proposition 5.4.2. **Right plot.** Compression ($\sigma \equiv \mathrm{sign}$) of the grayscale CIFAR-10 "cat" class with a two-layer autoencoder. The data is *not whitened* ($\boldsymbol{\Sigma} \neq \boldsymbol{I}$). The blue dots are the SGD population risk, and they are close to the lower bound of Theorem 8.

with a polynomial number of neurons [Ngu21]), see Section 5.2. In contrast, in this paper we consider *non-linear* autoencoders trained in the *challenging proportional regime*, in which the number of inputs to compress scales linearly with the size of the representation. More specifically, we consider the prototypical model of a two-layer autoencoder

$$\hat{\boldsymbol{x}}(\boldsymbol{x}) := \hat{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{A}, \boldsymbol{B}) = \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x}). \tag{5.1}$$

Here, $\boldsymbol{x} \in \mathbb{R}^d$ is the input to compress, $\hat{\boldsymbol{x}} \in \mathbb{R}^n$ the reconstruction, $\boldsymbol{B} \in \mathbb{R}^{n \times d}$ the encoding matrix, and $\boldsymbol{A} \in \mathbb{R}^{d \times n}$ the decoding matrix; the activation $\sigma : \mathbb{R} \to \mathbb{R}$ is applied *element-wise*. We aim at minimizing the population risk

$$\mathcal{R}(\boldsymbol{A}, \boldsymbol{B}) := d^{-1}\mathbb{E}_{\boldsymbol{x}} \left\| \boldsymbol{x} - \hat{\boldsymbol{x}}(\boldsymbol{x}) \right\|_2^2, \tag{5.2}$$

where the expectation is taken over the distribution of the input $\boldsymbol{x}$. Our focus is on Gaussian input data, i.e., $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$. When σ is the sign function, the encoder $\sigma(\boldsymbol{B}\boldsymbol{x})$ can be interpreted as a *compressor*, namely, it compresses the d-dimensional input signal into n bits. The problem (5.2) of compressing a Gaussian source with quadratic distortion has been studied in exquisite detail in the information theory literature [CT06], and the optimal performance for general encoder/decoder pairs is known via the *rate-distortion* formalism which characterizes the lowest achievable distortion in terms of the rate $r = n/d$. Here, we focus on encoders and decoders that form the two-layer autoencoder (5.1): we study the fundamental limits of this learning problem, as well as the performance achieved by commonly used gradient descent methods.

Main contributions. Taken all together, our results show that, for two-layer autoencoders, gradient descent methods achieve a global minimizer of the population risk: this is rigorously proved in the isotropic case ($\boldsymbol{\Sigma} = \boldsymbol{I}$) and corroborated by numerical simulations for a general covariance $\boldsymbol{\Sigma}$. Furthermore, we unveil the structure of said minimizer: for $\boldsymbol{\Sigma} = \boldsymbol{I}$, the optimal decoder has unit singular values; for general covariance, the spectrum of the decoder exhibits the same block structure as $\boldsymbol{\Sigma}$, and it can be explicitly obtained from $\boldsymbol{\Sigma}$ via a water-filling criterion; in all cases, weight-tying is optimal, i.e., $\boldsymbol{A}$ is proportional to $\boldsymbol{B}^\top$. Specifically, our technical results can be summarized as follows.

- Section 5.4.1 characterizes the minimizers of the risk (5.2) for isotropic data: Theorem 5 provides a tight lower bound, which is achieved by the set (5.7) of weight-tied *orthogonal* matrices, when the compression rate $r = n/d \leq 1$; for $r > 1$, Propositions 5.4.2 and 5.4.3 give a lower bound, which is approached (as $d \to \infty$) by the set (5.12) of weight-tied *rotationally invariant* matrices.

- Section 5.4.2 shows that the above minimizers are reached by gradient descent methods for $r \leq 1$: Theorem 6 shows linear convergence of *gradient flow* for general initializations, under a weight-tying condition; Theorem 7 considers a Gaussian initialization and proves global convergence of the *projected gradient descent* algorithm, in which the encoder matrix $\boldsymbol{B}$ is optimized via a gradient method and the decoder matrix $\boldsymbol{A}$ is obtained directly via linear regression.
- Section 5.5 focuses on data with general covariance $\boldsymbol{\Sigma} \neq \boldsymbol{I}$. We observe that experimentally weight-tying is optimal and then derive the corresponding lower bound (see Theorem 8), which is also asymptotically achieved (as $d \to \infty$) by rotationally invariant matrices with a carefully designed spectrum (depending on $\boldsymbol{\Sigma}$), see Proposition 5.5.2.

When $\sigma \equiv \mathrm{sign}$, our analysis characterizes the fundamental limits of the lossy compression of a Gaussian source via two-layer autoencoders. Remarkably, if we restrict to a certain class of *linear encoders* for compression, two-layer autoencoders achieve optimal performance [TCVS13], which can be generally obtained via a message passing decoding algorithm [RSF19]. However, for *general encoder/decoder pairs*, shallow autoencoders fail to meet the information-theoretic bound given by the rate-distortion curve, see Section 5.6.

Going beyond the Gaussian assumption on the data, we provide numerical validation to our theoretical predictions on standard datasets, both in the isotropic case and for general covariance (Figure 5.1). Additional numerical results – together with the details of the experimental setting – are in Appendix C.7.

Proof techniques. The lower bound on the population risk of Theorem 5 comes from a sequence of relaxations of the objective function, which eventually allows to apply a trace inequality. For $r \geq 1$, Proposition 5.4.2 crucially exploits an inequality for the Hadamard product of PSD matrices [Kha21], and the asymptotic achievability of Proposition 5.4.3 takes advantage of concentration-of-measure tools for orthogonal matrices. The key quantity in the analysis of gradient methods is the encoder Gram matrix at iteration t, i.e., $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$. In particular, for gradient flow (Theorem 6), due to the weight-tying condition, tracking $\log\det \boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ leads to a quantitative convergence result. However, when the weights are not tied, this quantity does not appear to increase along the optimization trajectory. Thus, for projected gradient descent (Theorem 7), the idea is to decompose $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ into *(i)* its value at the optimum (given by the identity), *(ii)* the contribution due to the spectrum evolution (keeping the eigenbasis fixed), and *(iii)* the change in the eigenbasis. Via a sequence of careful approximations, we are able to show that the term *(iii)* vanishes. Hence, we can study explicitly the evolution of the spectrum and obtain the desired convergence.

5.2 Related Work

Theory of autoencoders. A popular line of work has focused on two-layer *linear* autoencoders: [OSWS20] analyze the loss landscape; [KBGS19] show that the minimizers of the regularized loss recover the principal components of the data and, notably, the corresponding autoencoder is weight-tied; [BLSG20] prove that stochastic gradient descent – after a slight perturbation – escapes the saddles and eventually converges; [GBLJ19] characterize the time-steps at which the network learns different sets of features. [RMB$^+$18, NWH19] prove local convergence for weight-tied two-layer ReLU autoencoders. [NWH21] focus on the lazy training regime [COB19, JGH18] and bound the over-parameterization needed for global convergence. [RBU20] show that over-parameterized autoencoders learn solutions that are contractive around the training examples. The latent spaces of autoencoders are studied in [JRU21], where it is shown that such latent spaces can be aligned by stretching along the left singular vectors of the data. More closely related to our work, [Ngu21] and [RG22] track the gradient dynamics of *non-linear* two-layer autoencoders via the mean-field PDE and a system of ODEs, respectively. However, these analyses are restricted to diverging and vanishing rates: [Ngu21] considers weight-tied autoencoders with polynomially many neurons in the input dimension (so that $r \to \infty$); [RG22] consider the other extreme regime in which the input dimension diverges (so that $r \to 0$).

Neural compression. In recent years, compressors based on neural networks have outperformed traditional schemes on real-world data in terms of minimizing distortion and producing visually pleasing reconstructions at reasonable complexity [BLS17, TSCH17, AMT$^+$17, BCM$^+$21]. These methods typically use an autoencoder architecture with quantization of the latent variables, which is trained over samples drawn from the source. More recently, other architectures such as attention or diffusion-based models have been incorporated into neural compressors [CSTK20, LCG$^+$19, YM23, TSHM22], and improvements have been observed. We refer to [YMT22] for a detailed review on this topic. Given the remarkable success of neural compressors, it is imperative to understand the fundamental limits of compression using neural architectures. In this regard, [WB21] consider a highly-structured and low-dimensional random process, dubbed the *sawbridge*, and show numerically that the rate-distortion function is achieved by a compressor based on deep neural networks trained via stochastic gradient descent. In contrast, our work considers Gaussian sources, which are high-dimensional in nature, and provides the fundamental limits of compression for two-layer autoencoders. Our results also imply that two-layer autoencoders cannot achieve the rate-distortion limit on Gaussian data, see Section 5.6.

Rate-distortion formalism. Lossy compression of stationary sources is a classical problem in information theory, and several approaches have been proposed, including vector quantization [Gra84], or the usage of powerful channel codes [KU10, CMZ06, WMM10]. The rate-distortion function characterizes the optimal trade-off between error and size of the representation for the compression of an i.i.d. source [Sha48, Sha59, CT06]. However, computing the rate-distortion function is by itself a challenging task. The Blahut-Arimoto scheme [Bla72, Ari72] provides a systematic approach, but it suffers from the issue of scalability [LHB22]. Consequently, to compute the rate-distortion of empirical datasets, approximate methods based on generative modeling have been proposed [YM21b, LHB22].

Non-linear inverse problems. The task of estimating a signal $\boldsymbol{x}$ from non-linear measurements $\boldsymbol{y} = \sigma(\boldsymbol{B}\boldsymbol{x})$ has appeared in many areas, such as 1-bit compressed sensing where $\sigma(z) = \text{sign}(z)$ [BB08], or phase retrieval where $\sigma(z) = |z|$ [CSV13, CLS15]. While the focus of these problems is different from ours (e.g., compressed sensing has often an additional sparsity assumption), the ideas and proof techniques developed in this paper might be beneficial to characterize the fundamental limits and the performance of gradient-based methods for general inverse reconstruction tasks, see e.g. [MXM21, MM22].

5.3 Preliminaries

Notations. We use plain symbols for real numbers (e.g., a, b), bold symbols for vectors (e.g., $\boldsymbol{a}, \boldsymbol{b}$), and capitalized bold symbols for matrices (e.g., $\boldsymbol{A}, \boldsymbol{B}$). We let $[n] = \{1, \dots, n\}$, $\boldsymbol{I}$ be the identity matrix and $\mathbf{1}$ the column vector containing ones. Given a matrix $\boldsymbol{A}$, we denote its operator norm by $\|\boldsymbol{A}\|_{op}$ and its Frobenius norm by $\|\boldsymbol{A}\|_F$. Given two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ of the same shape, we denote their element-wise (Hadamard/Schur) product by $\boldsymbol{A} \circ \boldsymbol{B}$ and the k-th element-wise power by $\boldsymbol{A}^{\circ k}$. We write $L^2(\mathbb{R}, \mu)$ for the space of L^2 integrable functions on $\mathbb{R}$ w.r.t. the standard Gaussian measure μ and $h_k(x)$ for the k-th normalized Hermite polynomial (see e.g. [O'D14]).

Setup. We consider the two-layer autoencoder (5.1) and aim at minimizing the population risk (5.2) for a given rate $r = n/d$. In particular, we provide tight lower bounds on the minimum of the population risk computed on Gaussian input data with covariance $\boldsymbol{\Sigma}$, i.e.,

$$\widehat{\mathcal{R}}(r) := \min_{\boldsymbol{A},\boldsymbol{B}} \mathcal{R}(\boldsymbol{A}, \boldsymbol{B}). \tag{5.3}$$

In the isotropic case ($\boldsymbol{\Sigma} = \boldsymbol{I}$), our results hold for any odd activation $\sigma \in L^2(\mathbb{R}, \mu)$ after restricting the rows of the encoding matrix $\boldsymbol{B}$ to have unit norm. We remark that, when $\sigma(x) = \operatorname{sign}(x)$, the restriction is unnecessary since the activation is homogeneous.[1] We also note that restricting the norms of the rows of $\boldsymbol{B}$ prevents the model from entering the "linear" regime. In fact, when $\|\boldsymbol{B}\|_F \approx 0$, by linearizing the activation around zero, (5.1) reduces to the linear model $\hat{\boldsymbol{x}}(\boldsymbol{x}) \approx \boldsymbol{A}\boldsymbol{B}\boldsymbol{x}$, which exhibits a PCA-like behaviour. For general covariance $\boldsymbol{\Sigma}$, we consider odd homogeneous activations, which includes the sign function and monomials of arbitrary odd degree.

Any function $\sigma \in L^2(\mathbb{R}, \mu)$ can be expanded in terms of Hermite polynomials. This allows to perform Fourier analysis in the Gaussian space $L^2(\mathbb{R}, \mu)$, and it provides a natural tool because of the Gaussian assumption on the data. In particular, for odd σ, only odd Hermite polynomials occur, i.e.,

$$\sigma(x) = \textstyle\sum_{\ell=0}^{\infty} c_{2\ell+1} h_{2\ell+1}(x), \tag{5.4}$$

where $\{c_\ell\}_{\ell \in \mathbb{N}}$ denote the Hermite coefficients of σ. We also consider the following auxiliary quantity

$$\widetilde{\mathcal{R}}(r) := \min_{\boldsymbol{A}, \|(\boldsymbol{B}\boldsymbol{D})_{i,:}\|_2 = 1} \mathcal{R}(\boldsymbol{A}, \boldsymbol{B}), \tag{5.5}$$

that defines a minimum of the population risk for the autoencoder (5.1) with a certain norm constraint on the encoder weights $\boldsymbol{B}$. Here, $\boldsymbol{D}$ contains the square roots of the eigenvalues of $\boldsymbol{\Sigma}$ (i.e., $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{U}^\top$ for an orthogonal matrix $\boldsymbol{U}$), and $(\boldsymbol{B}\boldsymbol{D})_{i,:}$ stands for the i-th row of the matrix $\boldsymbol{B}\boldsymbol{D}$. A few remarks about the restricted population risk (5.5) are in order. First of all, if σ is homogeneous, the minimum of the restricted population risk (5.5) and of the unconstrained one (5.3) coincide (see Lemma 5.4.1 and Lemma 5.5.1). Thus, in this case, the analysis of $\widetilde{R}(r)$ directly provides results on the quantity of interest, i.e., $\widehat{\mathcal{R}}(r)$. The technical advantage of analysing (5.5) over (5.3) comes from fact that the expectation with respect to the Gaussian inputs, which arises in the constrained objective, can be explicitly computed via the reproducing property of Hermite polynomials (see, e.g., [O'D14]). To exploit this reproducing property, it is crucial that the inner products $\langle \boldsymbol{B}_{i,:}, \boldsymbol{x} \rangle$ have the same scale, which is ensured by picking $\|(\boldsymbol{B}\boldsymbol{D})_{i,:}\|_2 = 1$. The sole dependence of the constraint on the spectrum $\boldsymbol{D}$ (and, not on a particular choice of $\boldsymbol{U}$) stems from the rotational invariance of the isotropic Gaussian distribution.

[1] We say that a function σ is homogeneous if there exists an integer k s.t. $\sigma(\alpha x) = \alpha^k \sigma(x)$ for all $\alpha \neq 0$.

5.4 Main Results

In this section, we consider isotropic Gaussian data, i.e., $\boldsymbol{\Sigma} = \boldsymbol{D} = \boldsymbol{I}$. First, we derive a closed form expression for the population risk in Lemma 5.4.1. Then, in Theorem 5 we give a lower bound on the population risk for $r \leq 1$ and provide a complete characterization of the autoencoder parameters $(\boldsymbol{A}, \boldsymbol{B})$ achieving it. Surprisingly, the minimizer exhibits a *weight-tying* structure and the corresponding matrices are *rotationally invariant*. Later, in Proposition 5.4.2 we derive an analogous lower bound for $r > 1$. While it is hard to characterize the minimizer structure explicitly for a finite input dimension d (and $r > 1$), we provide a sequence $\{(\boldsymbol{A}_d, \boldsymbol{B}_d)\}_{d\in\mathbb{N}}$ that meets the lower bound in the high-dimensional limit ($d \to \infty$), see Proposition 5.4.3. Notably, the elements of this sequence share the key features (weight-tying, rotational invariance) of the minimizers for $r \leq 1$. In Section 5.4.2 we describe gradient methods that provably achieve the optimal value of the population risk. Specifically, we consider gradient flow under a weight-tying constraint and projected (on the sphere) gradient descent with Gaussian initialization. The corresponding results are stated in Theorem 6 and Theorem 7.

We start by expanding σ in a Hermite series to obtain a closed-form expression for the population risk.

Lemma 5.4.1. *Consider any odd $\sigma \in L^2(\mathbb{R}, \mu)$ and its Hermite expansion given by* (5.4)*. Then, $\widetilde{\mathcal{R}}(r)$ is equivalent to*

$$\min_{\boldsymbol{A}, \|\boldsymbol{B}_{i,:}\|_2=1} \frac{1}{d}\left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right]\right) + 1, \tag{5.6}$$

where $f(x) := \sum_{\ell=0}^{\infty}(c_{2\ell+1})^2 x^{2\ell+1}$ is applied element-wise. In particular, if $\sigma(x) = \mathrm{sign}(x)$, then $f(x) = c_1^2 \cdot \arcsin(x)$ and $c_1 = \sqrt{2/\pi}$. Moreover, for any homogeneous σ, we have that $\widehat{\mathcal{R}}(r) = \widetilde{\mathcal{R}}(r)$.

The proof of the lemma above is contained in Appendix C.1. Note that, if $c_1 = 0$, it is easy to see that the minimum of $\widetilde{R}(r)$ equals 1 and it is attained when $\boldsymbol{A}^\top\boldsymbol{A}$ is the zero-matrix. Furthermore, if $\sum_{\ell=1}^{\infty}(c_{2\ell+1})^2 = 0$, then $\sigma(x) = c_1^2 x$ and we fall back into the simpler case of a linear autoencoder [BH89, KBGS19, GBLJ19]. Thus, for the rest of the section, we will assume that $c_1 \neq 0$ and $\sum_{\ell=1}^{\infty}(c_{2\ell+1})^2 \neq 0$.

5.4.1 Fundamental Limits: Lower Bound on Risk

We begin by providing a tight lower bound for $r \leq 1$, which is *uniquely* achieved on the set of *weight-tied* orthogonal matrices $\mathcal{H}_{n,d}$ defined as

$$\mathcal{H}_{n,d} := \left\{\widetilde{\boldsymbol{A}}, \widetilde{\boldsymbol{B}}^\top \in \mathbb{R}^{d\times n} : \widetilde{\boldsymbol{A}} = \frac{c_1}{f(1)} \cdot \widetilde{\boldsymbol{B}}^\top, \widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{B}}^\top = \boldsymbol{I}\right\}. \tag{5.7}$$

Theorem 5. *Consider any odd $\sigma \in L^2(\mathbb{R}, \mu)$ and fix $r \leq 1$. Then, the following holds*

$$\widetilde{\mathcal{R}}(r) \geq \mathrm{LB}_{r\leq 1}(\boldsymbol{I}) := 1 - \frac{c_1^2}{f(1)} \cdot r,$$

and equality is achieved iff $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{H}_{n,d}$.

We note that the minimizers $\mathcal{H}_{n,d}$ of $\widetilde{\mathcal{R}}(r)$ do not directly correspond to the minimizers of the unconstrained population risk $\widehat{\mathcal{R}}(r)$, since in general $\widetilde{\mathcal{R}}(r) \neq \widehat{\mathcal{R}}(r)$. However, if σ is homogeneous, the "inverse" mapping can be readily obtained. For instance, when $\sigma(x) = \mathrm{sign}(x)$, rescaling the norms of the rows of $\boldsymbol{B}$ does not affect the compression, i.e., $\mathrm{sign}(\boldsymbol{Bx}) = \mathrm{sign}(\boldsymbol{SBx})$ for any diagonal $\boldsymbol{S}$ with positive entries. Hence, to obtain a minimizer, it suffices that the rows of $\boldsymbol{B}$ form any set of orthogonal (not necessarily normalized) vectors. In contrast, note that $\boldsymbol{A}$ is still defined with respect to the row-normalized version of $\boldsymbol{B}$. Similar arguments hold for homogeneous activations.

We also note that the weight-tying structure (5.7) observed in the minimizers of the population risk is related to the early representation learning literature [VLBM08, HS06].

It is also worth noting that the optimal value of the population risk in Theorem 5 is attained by a simple quantization scheme that processes the coordinates of the d-dimensional input $\boldsymbol{x}$ *independently*. In particular, it means that the performance of a two-layer model (5.1) is far from being optimal (see discussion in Section 5.6 and Figure 5.3), since the optimal scheme will take advantage of compressing the coordinates of the inputs *jointly*. We now describe the quantization scheme mentioned before. In a nutshell, each of the input's coordinates is mapped to the particular value that depends on the sign of the coordinate, namely:

$$q_a(x_i) := a \cdot \mathrm{sign}(x_i), \quad i \in [d],$$

for some positive quantization center a. In order to pick the optimal center a^*, we minimize the MSE directly, i.e.,

$$a^* = \arg\min_a \mathbb{E}\left[(x_i - q_a(x_i))^2\right],$$

which gives $a^* = \sqrt{2/\pi}$. It is then a simple substitution to check that the corresponding population risk will exactly coincide with the value in Theorem 5 (since $r < 1$ the coordinates that do not "fit" in the hidden representation are set to 0).

We now provide a proof sketch for Theorem 5 and defer the full argument to Appendix C.2.1.

Proof sketch of Theorem 5. Using the series expansion of $f(\cdot)$, we can write

$$\begin{aligned}&\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2c_1 \cdot \mathrm{Tr}\left[\boldsymbol{BA}\right] \\ &= \sum_{\ell=0}^{\infty} c_{2\ell+1}^2 \left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A}\left(\boldsymbol{B}\boldsymbol{B}^\top\right)^{\circ 2\ell+1}\right] - 2\frac{c_1}{f(1)}\mathrm{Tr}\left[\boldsymbol{BA}\right]\right).\end{aligned}$$

Thus, the minimization problem in Lemma 5.4.1 can be reduced to analysing each Hadamard power individually:

$$\min_{\boldsymbol{A}, \|\boldsymbol{B}_{i,:}\|_2 = 1} \mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A}(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ \ell}\right] - \frac{2c_1}{f(1)} \cdot \mathrm{Tr}\left[\boldsymbol{BA}\right]. \tag{5.8}$$

The crux of the argument is to provide a suitable sequence of relaxations of (5.8). The first relaxation gives

$$\mathrm{Tr}\left[(\boldsymbol{A}^\top \boldsymbol{A} \circ \boldsymbol{Q})(\boldsymbol{B}\boldsymbol{B}^\top \circ \boldsymbol{Q})\right] - \frac{2c_1}{f(1)} \cdot \mathrm{Tr}\left[\boldsymbol{BA}\right], \tag{5.9}$$

where $\boldsymbol{Q}$ is *any* PSD matrix with unit diagonal. Using the properties of the SVD of $\boldsymbol{Q}$, (5.9) can be further relaxed to

$$\sum_{i,j=1}^{n} \mathrm{Tr}\left[\boldsymbol{A}_j \boldsymbol{A}_j^\top \boldsymbol{B}_j \boldsymbol{B}_j^\top\right] - \frac{2c_1}{f(1)} \cdot \sum_{i=1}^{n} \mathrm{Tr}\left[\boldsymbol{B}_i \boldsymbol{A}_i\right], \tag{5.10}$$

where now $\boldsymbol{A}_i, \boldsymbol{B}_i^\top \in \mathbb{R}^{d\times n}$ are arbitrary matrices. The key observation is that

$$\sum_{i=1}^{n} \left\| \frac{c_1}{f(1)} \cdot \sqrt{\boldsymbol{X}}^{-1} \boldsymbol{A}_i^\top - \sqrt{\boldsymbol{X}} \boldsymbol{B}_i \right\|_F^2 = (5.10) + \frac{c_1^2}{(f(1))^2} \cdot n,$$

with $\boldsymbol{X} = \sum_{i=1}^n \boldsymbol{A}_i^\top \boldsymbol{A}_i$. As each relaxation lower bounds (5.8) and the Frobenius norm is positive, this argument leads to the lower bound on $\widetilde{R}(r)$. The fact that the lower bound is met for any $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{H}_{n,d}$ can be verified via a direct calculation. The uniqueness follows by taking the intersection of the minimizers of (5.8) for different values of ℓ. □

Next, we move to the case $r > 1$.

Proposition 5.4.2. *Consider any odd $\sigma \in L^2(\mathbb{R}, \mu)$ and fix $r > 1$, then the following holds:*

$$\widetilde{\mathcal{R}}(r) \geq \mathrm{LB}_{r>1}(\boldsymbol{I}) := 1 - \frac{r}{r + \left(\frac{f(1)}{c_1^2} - 1\right)}.$$

The key difference with the proof of the lower bound in Theorem 5 is that the term $\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \boldsymbol{B} \boldsymbol{B}^\top\right]$ requires a tighter estimate. This is due to the fact that the matrix $\boldsymbol{B}\boldsymbol{B}^\top$ is no longer full-rank when $r > 1$. We obtain the desired tighter bound by exploiting the following result by [Kha21]:

$$\boldsymbol{A}^\top \boldsymbol{A} \circ \boldsymbol{B}\boldsymbol{B}^\top \succeq \frac{1}{d} \cdot \mathrm{Diag}(\boldsymbol{B}\boldsymbol{A}) \mathrm{Diag}(\boldsymbol{B}\boldsymbol{A})^\top, \tag{5.11}$$

where $\mathrm{Diag}(\boldsymbol{B}\boldsymbol{A})$ stands for the vector containing the diagonal entries of $\boldsymbol{B}\boldsymbol{A}$. The full argument is contained in Appendix C.2.2.

As for $r \leq 1$, the bound is met (here, in the limit $d \to \infty$) by considering weight-tied matrices:

$$\hat{\boldsymbol{B}}^\top = \sqrt{r} \cdot [\boldsymbol{I}_d, \boldsymbol{0}_{d,n-d}] \boldsymbol{U}^\top, \ \boldsymbol{b}_i = \frac{\hat{\boldsymbol{b}}_i}{\|\hat{\boldsymbol{b}}_i\|_2}, \ \boldsymbol{A} = \beta \boldsymbol{B}^\top, \tag{5.12}$$

where $\beta = \frac{c_1}{c_1^2 r + f(1) - c_1^2}$ and $\boldsymbol{U}$ is uniformly sampled from the group of rotation matrices. The idea behind the choice (5.12) is that, as $d \to \infty$, $(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ 2\ell}$ for $\ell \geq 2$ is close to the identity matrix, and (5.11) is attained exactly. The formal statement is provided below and proved in Appendix C.2.2.

Proposition 5.4.3. *Consider any odd $\sigma \in L^2(\mathbb{R}, \mu)$ and fix $r > 1$. Let $\boldsymbol{A}, \boldsymbol{B}$ be defined as in* (5.12)*. Then, for any $\epsilon > 0$ the following holds*

$$|\mathcal{R}(\boldsymbol{A}, \boldsymbol{B}) - \mathrm{LB}_{r>1}(\boldsymbol{I})| \leq C d^{-\frac{1}{2}+\epsilon},$$

with probability $1 - c/d^2$. Here, the constants c, C depend only on r and ϵ.

Degenerate isotropic Gaussian data. All the arguments of Section 5.4.1 directly apply for $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$, the only differences being the scaling of the term $\mathrm{Tr}[\boldsymbol{B}\boldsymbol{A}]$ (which is additionally multiplied by σ) and the constant variance term σ^2 (in place of 1) in (5.6). Our results can be also easily extended to the case of degenerate isotropic Gaussian data, i.e., $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ with $\lambda_i(\boldsymbol{\Sigma}) = \sigma^2$ for $i \leq d - k$ and $\lambda_i(\boldsymbol{\Sigma}) = 0$ for $i > d - k$, where $\lambda_i(\boldsymbol{\Sigma})$ stands for the i-th eigenvalue of $\boldsymbol{\Sigma}$ in non-increasing order. In fact, by the rotational invariance of the Gaussian distribution, we can assume without loss of generality that $\boldsymbol{x} = [x_1, \cdots, x_{d-k}, 0, \cdots, 0]$, where $(x_i) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$. Hence, by considering $\boldsymbol{A} \in \mathbb{R}^{(d-k)\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{n \times (d-k)}$ and substituting d with $d - k$ where suitable, analogous results follow.

5.4.2 Gradient Methods Achieve the Lower Bound

In this section, we discuss the achievability of the lower bound obtained in the previous section via gradient methods. We study two procedures which find the minimizer of the population risk $\mathcal{R}(\boldsymbol{A}, \boldsymbol{B})$ under the constraint $\|\boldsymbol{B}_{i,:}\|_2 = 1$. Namely, we analyse *(i) weight-tied gradient flow* on the sphere and *(ii)* its discrete version (with finite step size) *without* weight-tying, i.e., *projected gradient descent*.

The optimization objective in Lemma 5.4.1 is equivalent (up to a scaling independent of $(\boldsymbol{A}, \boldsymbol{B})$) to

$$\min_{\boldsymbol{A}, \|\boldsymbol{B}_{i,:}\|_2=1} \operatorname{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2 \cdot \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right], \tag{5.13}$$

where we have rescaled the function f by $1/c_1^2$. This follows from the fact that the multiplicative factor c_1 can be pushed inside $\boldsymbol{A}$. Note that such scaling does not affect the properties of gradient-based algorithms (modulo a constant change in their speed). Hence, without loss of generality, we will state and prove all our results for the problem (5.13).

Weight-tied gradient flow. We start with the weight-tied setting, in which

$$\boldsymbol{A} = \beta \boldsymbol{B}^\top, \quad \beta \in \mathbb{R}. \tag{5.14}$$

This is motivated by the fact that the lower bounds on the population risk are approached by weight-tied matrices (see Theorem 5 and Proposition 5.4.3). Under the weight-tying constraint (5.14), the objective (5.13) has the following form

$$\begin{aligned}\Psi(\beta, \boldsymbol{B}) &:= \beta^2 \cdot \operatorname{Tr}\left[\boldsymbol{B}^\top \boldsymbol{B} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2\beta n \\ &= \beta^2 \cdot \sum_{i,j=1}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right) - 2\beta n,\end{aligned} \tag{5.15}$$

where $\|\boldsymbol{b}_i\|_2 = 1$ for all i. Note that the optimal β^* can be found exactly, since (5.15) is a quadratic polynomial in β. In this view, to optimize (5.15), we perform a gradient flow on $\{\boldsymbol{b}_i\}_{i=1}^n$, which are regarded as vectors on the unit sphere, and pick the optimal β^* at each time t. Formally,

$$\begin{aligned}\beta(t) &= \frac{n}{\sum_{i,j=1}^n \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right)}, \\ \frac{\partial \boldsymbol{b}_i(t)}{\partial t} &= -\boldsymbol{J}_i(t) \nabla_{\boldsymbol{b}_i} \Psi(\beta(t), \boldsymbol{B}(t)),\end{aligned} \tag{5.16}$$

where $\boldsymbol{J}_i(t) := \boldsymbol{I} - \boldsymbol{b}_i(t)\boldsymbol{b}_i(t)^\top$ projects the gradient $\nabla_{\boldsymbol{b}_i} \Psi(\beta(t), \boldsymbol{B}(t))$ onto the tangent space at the point $\boldsymbol{b}_i(t)$ (see (C.45) in Appendix C.3 for the closed form expression). This ensures that $\|\boldsymbol{b}_i(t)\|_2 = 1$ along the gradient flow trajectory. The described procedure can be viewed as Riemannian gradient flow, due to the projection of the gradient $\nabla_{\boldsymbol{b}_i} \Psi(\beta(t), \boldsymbol{B}(t))$ on the tangent space of the unit sphere.

Theorem 6. *Fix $r \leq 1$. Let $\boldsymbol{B}(t)$ be obtained via the gradient flow* (5.16) *applied to Ψ defined in* (5.15)*. Let the initialization $\boldsymbol{B}(0)$ have unit-norm rows and $\operatorname{rank}(\boldsymbol{B}(0)) = n$. Then, as $t \to \infty$, $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ converges to $\boldsymbol{I}$, which is the unique global optimum of* (5.15)*. Moreover, define the residual*

$$\phi(t) = \operatorname{Tr}\left[(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top - \boldsymbol{I}) \cdot f(\boldsymbol{B}(t)^\top \boldsymbol{B}(t))\right] \geq 0, \tag{5.17}$$

which vanishes at the minimizer, and let T be the first time such that $\phi(T) = \delta$. Then,

$$\begin{aligned} T \leq &- \mathbb{1}\{\phi(0) > nf(1)\} \cdot f(1) \cdot \log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) \\ &- \mathbb{1}\{\delta \leq nf(1)\} \cdot \frac{2f^2(1)}{\delta} \cdot \log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top). \end{aligned} \tag{5.18}$$

In words, if the residual at initialization is bigger than $nf(1)$, then it takes at most constant time to reach the regime in which the convergence is linear in the precision δ. We also note that by choosing the optimal β^*, the function ϕ can be related to the objective (5.15) by $\Psi(\beta^*, \boldsymbol{B}(t)) = -\frac{n}{f(1)+\frac{\phi(t)}{n}}$. Hence, (5.18) gives a quantitative convergence in terms of the objective function as well. We give a sketch of the argument below and defer the complete proof to Appendix C.3.

Proof sketch of Theorem 6. It can be readily shown that $\boldsymbol{B}\boldsymbol{B}^\top = \boldsymbol{I}$ is a minimizer of (5.15) and a stationary point of the gradient flow (5.16). However, if the gradient flow (5.16) ends up in points for which $\mathrm{rank}(\boldsymbol{B}) < n$, such subspaces are never escaped (see Lemma C.3.1) and the procedure fails to converge to the full-rank global minimizer. Thus, our strategy is to show that, if at initialization $\mathrm{rank}(\boldsymbol{B}) = n$, the gradient flow will never collapse to $\mathrm{rank}(\boldsymbol{B}) < n$. To do so, the key intuition is to track the quantity $\log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top)$ during training. In particular, we show in Lemma C.3.2 that

$$\frac{\partial \log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top)}{\partial t} \geq \phi(t) \geq 0. \tag{5.19}$$

The inequality (5.19) implies that the determinant is non-decreasing and, hence, the smallest eigenvalue of $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ is bounded away from 0 (uniformly in t), which gives the desired full-rank property. The convergence speed also follows from (5.19) by a careful integration in time (see Lemma C.3.3). □

We remark that Theorem 6 holds for any d and for all full-rank initializations.

Projected gradient descent. We now move to the setting where the encoder and decoder weights are not weight-tied. In this case, we consider the commonly used Gaussian initialization and prove a result for sufficiently large d. The Gaussian initialization allows us to relax the requirement on f: we only need $c_2 = 0$, as opposed to the previous assumption that $c_{2\ell} = 0$ for any $\ell \in \mathbb{N}$ (see the statement of Lemma 5.4.1). Specifically, we consider the following algorithm to minimize (5.13):

$$\begin{aligned} &\boldsymbol{A}(t) = \boldsymbol{B}(t)^\top \left(f(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top)\right)^{-1} \\ &\boldsymbol{B}'(t) := \boldsymbol{B}(t) - \eta\nabla_{\boldsymbol{B}(t)}, \quad \boldsymbol{B}(t+1) := \mathrm{proj}(\boldsymbol{B}'(t)), \end{aligned} \tag{5.20}$$

where $\boldsymbol{A}(t)$ is the optimal matrix for a fixed $\boldsymbol{B}(t)$ and $\nabla_{\boldsymbol{B}(t)}$ (see (C.62) in Appendix C.4) is the projected gradient of the objective (5.13) with respect to $\boldsymbol{B}(t)$. Furthermore, $\mathrm{proj}(\boldsymbol{B}'(t))$ rescales all the rows to have unit norm. It will become apparent from the proof of Theorem 7 that the inversion in the definition of $\boldsymbol{A}(t)$ is indeed well defined. We remark that (5.20) can be viewed as the discrete counterpart of the Riemannian gradient flow (5.16) (with the optimal $\boldsymbol{A}(t)$ in place of the weight-tying), where the application of $\mathrm{proj}(\cdot)$ keeps the rows of $\boldsymbol{B}(t)$ of unit norm. In the related literature, this procedure is often referred to as Riemannian gradient descent (see, e.g., [AMS09]). Alternatively, (5.20) may be viewed as coordinate descent [Wri15] on the objective (5.13), where the step in $\boldsymbol{A}$ is performed exactly.

Our main result is that the projected gradient descent (5.20) converges to the global optimum of (5.13) for large enough d (with high probability). We give a sketch of the argument and defer the complete proof to Appendix C.4.

Theorem 7. *Consider the projected gradient descent* (5.20) *applied to the objective* (5.13) *for any f of the form $f(x) = x + \sum_{\ell=3} c_\ell^2 x^\ell$, where $\sum_{\ell=3} c_\ell^2 < \infty$. Initialize the algorithm with $\boldsymbol{B}(0)$ equal to a row-normalized Gaussian, i.e., $\boldsymbol{B}'_{i,j}(0) \sim \mathcal{N}(0, 1/d)$, $\boldsymbol{B}(0) = \mathrm{proj}(\boldsymbol{B}'(0))$. Let the step size η be $\Theta(1/\sqrt{d})$. Then, for any $r < 1$ and sufficiently large d, with probability at least $1 - Ce^{-cd}$, we have that $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ converges to $\boldsymbol{I}$, which is the unique global optimum of* (5.13)*. Moreover, defining $t = T/\eta$, we have the following bound on the rate of convergence*

$$\left\|\boldsymbol{B}(t)\boldsymbol{B}(t)^\top - \boldsymbol{I}\right\|_{op} \leq C(1-c)^T,$$

where $C > 0$ and $c \in (0, 1]$ are universal constants depending only on r and f.

Proof sketch of Theorem 7. Let $\boldsymbol{B}(0)\boldsymbol{B}(0)^\top = \boldsymbol{U}\boldsymbol{\Lambda}(0)\boldsymbol{U}^\top$ be the singular value decomposition (SVD) of the encoder Gram matrix. Then, the idea is to decompose $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ at each step of the projected gradient descent dynamics as

$$\boldsymbol{B}(t)\boldsymbol{B}(t)^\top = \boldsymbol{I} + \boldsymbol{Z}(t) + \boldsymbol{X}(t), \tag{5.21}$$

where $\boldsymbol{Z}(t) = \boldsymbol{U}(\boldsymbol{\Lambda}(t) - \boldsymbol{I})\boldsymbol{U}^\top$. Here, $\boldsymbol{I}$ is the global optimum towards which we want to converge; $\boldsymbol{Z}(t)$ captures the evolution of the eigenvalues while keeping the eigenbasis fixed, as $\boldsymbol{U}$ comes from the SVD at initialization; and $\boldsymbol{X}(t)$ is the remaining error term capturing the change in the eigenbasis. The update on $\boldsymbol{\Lambda}(t)$ is given by $\boldsymbol{\Lambda}(t+1) = g(\boldsymbol{\Lambda}(t))$, where $g : \mathbb{R}^n \to \mathbb{R}^n$ admits an explicit expression. Hence, in light of this explicit expression, if we had $\boldsymbol{X}(t) \equiv 0$, then the desired convergence would follow from the analysis of the recursion for $\boldsymbol{\Lambda}(t)$ (see Lemma C.5.3).

The main technical difficulty lies in carefully controlling the error term $\boldsymbol{X}(t)$. In particular, we will show that $\boldsymbol{X}(t)$ decays for large enough d, which means that dynamics (5.21) is well approximated by $\boldsymbol{I} + \boldsymbol{Z}(t)$. The proof can be broken down in four steps. In the *first step*, we compute the leading order term of $\nabla_{\boldsymbol{B}(t)}$ (see Lemma C.4.1 and C.4.2). This simplifies the formula for $\nabla_{\boldsymbol{B}(t)}$, which can then be expressed as an explicit nonlinear function of $\boldsymbol{Z}(t)$ and $\boldsymbol{X}(t)$. In the *second step*, we perform a Taylor expansion of $\nabla_{\boldsymbol{B}(t)}$, seen as a matrix-valued function in $\boldsymbol{Z}(t)$ and $\boldsymbol{X}(t)$ (see Lemma C.4.3). The intuition for this expansion comes from the fact that $\boldsymbol{X}(t)$ is a small quantity, and also $\|\boldsymbol{Z}(t)\|_{op} \to 0$ as $t \to \infty$. In the *third step*, we show that the norm of $\nabla_{\boldsymbol{B}(t)}$ vanishes sufficiently fast (see Lemma C.4.4), which implies that the projection step $\boldsymbol{B}(t+1) := \mathrm{proj}(\boldsymbol{B}'(t))$ has a negligible effect (see Lemma C.4.5). After doing these estimates, we finally obtain an explicit recursion for $\boldsymbol{X}(t)$. In the *fourth step*, we analyse this recursion (see Lemma C.4.6): first, we show that the error does not amplify too strongly (as in Gronwall's inequality); then, armed with this worst-case estimate, we can prove an exponential decay for $\boldsymbol{X}(t)$, which suffices to conclude the argument. □

Scaling of the learning rate. Theorem 7 is stated for $\eta = \Theta(1/\sqrt{d})$, as this corresponds to the biggest learning rate for which our argument works (thus requiring the least amount of steps for convergence). The same result can be proved for $\eta = \Theta(d^{-\kappa})$ with $\kappa \geq 1/2$. The only piece of the proof affected by this change is the third part of Lemma C.5.2 (in particular, the chain of inequalities (C.163)), which continues to hold as long as η is polynomial in d^{-1}.

Assumptions on compression rate r. We expect an analog of Theorem 6 to hold for $r > 1$, as long as d is *sufficiently large*. In fact, for a fixed d, it appears to be difficult to even characterize the global minimizer: the choice (5.12) approaches the lower bound $\mathrm{LB}_{r>1}(\boldsymbol{I})$ only as $d \to \infty$, see Proposition 5.4.3. We also expect Theorem 7 to hold for $r \geq 1$. Here, an additional challenge is that the minimizer has non-zero off-diagonal entries. In combination with the lack of an exact characterization of the minimizer, this leads to an additional error term that would be difficult to control with the current tools. At the same time, the restriction $r < 1$ is likely to be an artifact of the proof as experimentally (see, for instance, Figure 5.3) the algorithm still converges to the global optimum for $r \geq 1$.

Gaussian initialization in Theorem 7. The Gaussian initialization ensures that, with high probability, the off-diagonal entries of $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ are small. This allows us to approximate higher-order Hadamard powers of $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ with $\boldsymbol{I}$. However, in experiments the Gaussian assumption seems to be unnecessary, and we expect the convergence result to hold for all (non-degenerate) initializations.

5.5 Extension to General Covariance

In this section, we consider a Gaussian source with general covariance structure, i.e., $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{U}^\top$. Without loss of generality, the matrix $\boldsymbol{D}$ can be written as

$$\boldsymbol{D} = \mathrm{Diag}([\underbrace{D_1, \cdots, D_1}_{\times k_1} \,|\, \cdots \,|\, \underbrace{D_K, \cdots, D_K}_{\times k_K}]), \tag{5.22}$$

where $\sum_{i=1}^K k_i = d$, $k_i \geq 1$ and $D_i > D_{i+1} \geq 0$. We start by deriving a closed-form expression for the population risk, similar to that of Lemma 5.4.1. Its proof is given in Appendix C.1.

Lemma 5.5.1. *Let $\sigma \in L^2(\mathbb{R}, \mu)$ be an odd homogeneous activation, then $\widetilde{\mathcal{R}}(r)$ is equal to the minimum of*

$$\frac{1}{d}\left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{D}\boldsymbol{A}\right] + \mathrm{Tr}\left[\boldsymbol{D}^2\right]\right) \tag{5.23}$$

under the constraint $\|\boldsymbol{B}_{i,:}\|_2 = 1$. Moreover, $\widehat{\mathcal{R}}(r) = \widetilde{\mathcal{R}}(r)$.

The result of Lemma 5.5.1 can be extended to any odd $\sigma \in L^2(\mathbb{R}, \mu)$ at the cost of losing the equivalence between the objectives $\widehat{\mathcal{R}}(r)$ and $\widetilde{\mathcal{R}}(r)$.

We restrict the theoretical analysis to proving a lower bound on (5.23) in the weight-tied setting (5.14). This lower bound can be achieved via a careful choice of the matrices $\boldsymbol{A}, \boldsymbol{B}$ (see Proposition 5.5.2), and we provide numerical evidence (see Figure 5.2) that gradient descent saturates the bound *without* the weight-tying constraint. Thus, we expect our lower bound to hold also for general (not necessarily weight-tied) matrices.

The lower bound is given by the minimum

$$\frac{1}{d}\left(\frac{g(1)}{c_1^2 n}\left(\sum_{i=1}^K \beta_i\right)^2 + \sum_{i=1}^K \left(c_1^2 \frac{\beta_i^2}{s_i} - 2c_1 D_i \beta_i + D_i^2\right)\right) \tag{5.24}$$

over all $\beta_i \geq 0$ and

$$\begin{cases} 0 \leq s_i \leq \min\{k_i, n\}, \\ 1 \leq \sum_{i=1}^{K} s_i \leq \min\{d, n\}. \end{cases} \tag{5.25}$$

Here $g(x) = f(x) - c_1^2 x$, and we use the convention that $\frac{0^2}{0} = 0$ and $\frac{c}{0} = +\infty$ for $c > 0$. We can also explicitly characterize the optimal s_i, β_i. The optimal s_i are obtained via a *water-filling criterion*:

$$\begin{cases} \boldsymbol{s} = [n, 0, \cdots, 0], & n \leq k_1, \\ \boldsymbol{s} = [k_1, k_2, \cdots, k_K], & d \leq n, \\ \boldsymbol{s} = [k_1, \cdots, k_{\mathrm{id}(n)-1}, \mathrm{res}(n), 0, \cdots, 0] & \text{otherwise}, \end{cases} \tag{5.26}$$

where $\boldsymbol{s} = [s_1, \cdots, s_k]$, $\mathrm{id}(n)$ denotes the first position at which $\min\{n, d\} - \sum_{i=1}^{\mathrm{id}(n)} k_i < 0$, and the residual is defined by $\mathrm{res}(n) := \min\{n, d\} - \sum_{i=1}^{\mathrm{id}(n)-1} k_i$. The β_i can also be expressed explicitly in terms of f, s_i, D_i. This is summarized in the following theorem.

Theorem 8. *Consider the objective* (5.23) *under the weight-tied constraint* (5.14)*. Then,*

$$\text{(5.23)} \geq \mathrm{LB}(\boldsymbol{D}) := \min_{s_i, \beta_i} \text{(5.24)}, \tag{5.27}$$

where $\beta_i \geq 0$ and the s_i satisfy (5.25)*. Furthermore, the minimizers of* (5.24) *are the s_i obtained via the* water-filling *criterion* (5.26) *and*

$$\beta_i = \begin{cases} \frac{s_i}{c_1} \cdot \left(\frac{\frac{g(1)}{c_1^2 n} \sum_{j=1}^{M^*} s_j \Delta_j + D_1}{\frac{g(1)}{c_1^2 n} \sum_{j=1}^{M^*} s_j + 1} - \Delta_i \right) & \text{if } i \leq M^*, \\ 0 & \text{otherwise}, \end{cases} \tag{5.28}$$

where $\Delta_j = D_1 - D_j$ and M^* is smallest index such that

$$\frac{g(1)}{c_1^2 n} \sum_{j=1}^{M^*+1} s_j (D_{M^*+1} - D_j) + D_{M^*+1} \leq 0.$$

If no such index exists, then $M^* = K$.

We give a high-level overview of the proof below, and the complete argument is provided in Appendix C.6.

Proof sketch of Theorem 8. In the first step, we show that (5.27) holds. Consider the following block decomposition of $\boldsymbol{B}$ having the same block structure as $\boldsymbol{D}$:

$$\boldsymbol{B} = [\boldsymbol{\Gamma}_1 \boldsymbol{B}_1 | \cdots | \boldsymbol{\Gamma}_K \boldsymbol{B}_K], \tag{5.29}$$

where $\boldsymbol{B}_j \in \mathbb{R}^{n \times k_j}$ with $\|(\boldsymbol{B}_j)_{i,:}\|_2 = 1$ and $\{\boldsymbol{\Gamma}_j\}_{j=1}^K$ are diagonal matrices with $\sum_{j=1}^K \boldsymbol{\Gamma}_j^2 = \boldsymbol{I}$. Each $\boldsymbol{B}_i$ will play a similar role to the $\boldsymbol{B}$ in the isotropic case. The crucial bound for this step comes from Theorem A in [Kha21]:

$$(\boldsymbol{\Gamma}_i \boldsymbol{B}_i \boldsymbol{B}_i^\top \boldsymbol{\Gamma}_i)^{\circ 2} \succeq \frac{1}{s_i} \cdot \mathrm{Diag}(\boldsymbol{\Gamma}_i^2) \mathrm{Diag}(\boldsymbol{\Gamma}_i^2)^\top,$$

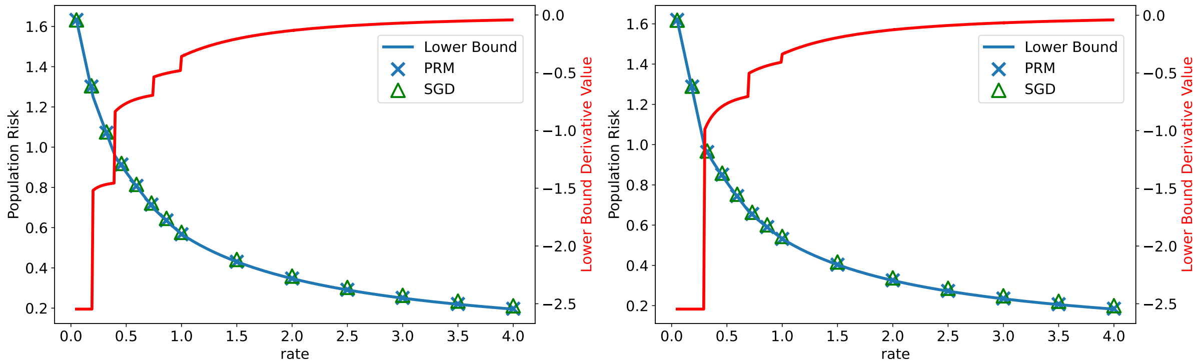


Figure 5.2: Compression ($\sigma \equiv \text{sign}$) of a non-isotropic Gaussian source, whose covariance matrix is obtained by taking $\boldsymbol{k} = (20, 20, 35, 25)$ and $(D_1, D_2, D_3, D_4) = (2, 1.5, 1, 0.8)$ for the left plot, and $\boldsymbol{k} = (30, 40, 30)$ and $(D_1, D_2, D_3) = (2, 1, 0.7)$ for the right plot. The blue crosses (Population Risk Minimizer, PRM) are obtained by optimizing (5.23) via GD. The green triangles are obtained by training an autoencoder via SGD on Gaussian samples with the given covariance structure. The red solid line plots the derivative of the population risk computed using a finite differences scheme. Note that the derivative jumps when the corresponding blocks are getting filled, although this may not happen in general, see Appendix C.7. A similar behavior can be observed in the isotropic case at $r = 1$, as there is only one block to fill (see Figure 5.3).

where $s_i = \text{rank}(\boldsymbol{B}_i \boldsymbol{B}_i^\top)$. Now, ignoring the (PSD) cross-terms for $i \neq j$ we can proceed as in the proof of Proposition 5.4.2 to arrive at the lower bound

$$\frac{1}{d}\left(\beta^2 \left(g(1) \cdot n + \sum_{i=1}^{K} \frac{\gamma_i^2}{s_i}\right) - 2\beta \cdot \sum_{i=1}^{K} D_i \gamma_i + \sum_{i=1}^{K} D_i^2\right), \tag{5.30}$$

where, with an abuse of notation, we have re-defined $g(x) := g(x)/c_1^2$ and $\beta := c_1 \beta$. Note that for $\boldsymbol{D} = \boldsymbol{I}$ one can easily find an expression for the minimum of (5.30) in terms of r and verify that it coincides with the previous bounds in Theorem 5 and Proposition 5.4.2. Now by choosing $\beta_i := \beta \gamma_i$ and using that $\sum_{i=1}^{K} \gamma_i = n$, the objective (5.30) is seen to be equivalent to (5.24), hence (5.27) holds.

Next, the optimal s_i are water-filled as defined in (5.26), which follows from the standard convex analysis argument of Lemma C.6.1. Finally, given the form of the optimal s_i, it remains to find the optimal β_i. This is done by considering a slightly more general problem in Lemma C.6.2. In fact, the problem of minimizing (5.24) is of the form:

$$(5.24) = \min_{m_i \geq 0} f\left(\sum_{i=1}^{K} m_i\right) + \sum_{i=1}^{K} f_i(m_i),$$

where importantly f and $\{f_i\}_{i=1}^K$ are *strictly convex* differentiable functions. The proof is based on techniques from convex analysis. The explicit calculations for our case are then carried out in Lemma C.6.3. □

Asymptotic achievability. We show that the lower bound in Theorem 8 can be asymptotically (i.e, as $d \to \infty$) achieved by using the block form (5.29), after carefully picking $\boldsymbol{B}_i$ for each block. Specifically, first we generate a matrix $\boldsymbol{U} \in \mathbb{R}^{n \times n}$ which is sampled uniformly from the group of orthogonal matrices. Next, we choose each $\boldsymbol{B}_i$ such that $\hat{\boldsymbol{B}}_i \hat{\boldsymbol{B}}_i^\top = \frac{n}{k_i} \boldsymbol{U} \boldsymbol{D}_i \boldsymbol{U}^\top$, where $\boldsymbol{D}_i$ is a diagonal matrix with

$$(\boldsymbol{D}_i)_{v,v} = \begin{cases} 1, & \text{if} \quad \sum_{j=1}^{i-1} k_j < v \leq \sum_{j=1}^{i} k_j, \\ 0, & \text{otherwise}, \end{cases}$$

and the rows of $\boldsymbol{B}$ are composed of normalized $\hat{\boldsymbol{b}}_i$, i.e., $\boldsymbol{b}_i = \frac{\hat{\boldsymbol{b}}_i}{\|\hat{\boldsymbol{b}}_i\|_2}$. Furthermore, we pick $\boldsymbol{\Gamma}_i^2 = \frac{\gamma_i}{n}\boldsymbol{I}$ and $\boldsymbol{A} = \beta\boldsymbol{B}^\top$. The scalings γ_i and β are chosen to be the minimizers of (5.30) for s_i as in (5.25). This is formalized in the following proposition.

Proposition 5.5.2. *Assume $\boldsymbol{A}, \boldsymbol{B}$ are constructed as described above and fix $r > 0$. Also assume that, for all i, $\frac{k_i}{n}$ converges to a strictly positive number as $d \to \infty$. Then, for any $\epsilon > 0$ with probability $1 - \frac{c}{d^2}$, the following holds*

$$|\mathcal{R}(\boldsymbol{A}, \boldsymbol{B}) - \mathrm{LB}(\boldsymbol{D})| \le Cd^{-\frac{1}{2}+\epsilon},$$

where $\mathrm{LB}(\boldsymbol{D})$ is defined in (5.27), and the constants c, C only depend on r, ϵ and $\lim_{d\to\infty} \frac{k_i}{n}$.

The proof of this lemma is similar to that of Proposition 5.4.3, and it is provided in Appendix C.6. We remark that Proposition 5.5.2 can be extended to D_i being sampled from a compactly supported measure, at the price of a worse rate of convergence. This is due to the fact that we can approximate compact measures with discrete measures. We omit the details here.

Taken together, Proposition 5.5.2 and Theorem 8 show that the optimal $\boldsymbol{B}$ exhibits the block structure (5.29), which agrees with the block structure (5.22) of the covariance matrix of the data. The individual blocks are orthogonal in the sense that $\boldsymbol{B}_i^\top \boldsymbol{\Gamma}_i \boldsymbol{\Gamma}_j \boldsymbol{B}_j = \boldsymbol{0}$. Furthermore, each block has the same form as the minimizers in the isotropic case, up to some scaling. Such a structure is also confirmed by the numerical experiments: for instance, it is observed in the settings considered for Figure 5.2.

5.6 Discussion

Population vs. empirical loss. All our results hold for the optimization of the population loss. Extending them to the empirical loss is an interesting direction for future research. One possible way forward is to exploit progress towards relating the landscape of empirical and population losses, see e.g. [MBM18]. We remark that, in the simulations of gradient descent, we always use the tempered straight-through estimator of the sign activation (see Appendix C.7 for details). Thus, another promising direction is to show that, in the low-temperature regime (i.e., when the differentiable approximation of the sign becomes almost perfect), the gradient-based scheme converges to the minimizer of the population risk.

Optimality of two-layer autoencoders. In this chapter we characterize the minimizers of the expected ℓ_2 error incurred by two-layer autoencoders, and show that the minimum error is achieved, under certain conditions, by gradient-based algorithms. Thus, for the special case in which $\sigma \equiv \mathrm{sign}$, a natural question is to what degree the model (5.1) is suitable for data compression. Let us fix the encoder to be a rotationally invariant matrix, i.e., $\boldsymbol{B} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^\top$ with $\boldsymbol{U}, \boldsymbol{V}$ independent and distributed according to the Haar measure and $\boldsymbol{\Lambda}$ having bounded entries. Then, the information-theoretically optimal reconstruction error can be computed via the replica method from statistical mechanics [TCVS13] and, in a number of scenarios, it coincides with the error of a Vector Approximate Message Passing (VAMP) algorithm [RSF19, SRF16]. Furthermore, it is also possible to optimize the spectrum $\boldsymbol{\Lambda}$ to minimize the error, which leads to the singular values of $\boldsymbol{B}$ being all 1 [MXM21].[2] Surprisingly,

[2] More specifically, [MXM21] consider an expectation propagation (EP) algorithm [Min01, OWJ05, FSARS16, HWJ17], which has been related to various forms of approximate message passing [MP17, RSF19].

for a compression rate $r \leq 1$, the optimal error found in [MXM21] *coincides* with the minimizer of the population loss given by Theorem 5. Hence, two-layer autoencoders are optimal compressors under two conditions: *(i)* $r \leq 1$, and *(ii)* fixed encoder given by a rotationally invariant matrix. Both conditions are sufficient and also necessary. For $r > 1$, VAMP outperforms the two-layer autoencoder. Moreover, for a *general encoder/decoder pair*, the information-theoretically optimal reconstruction error is given by the rate-distortion function, which outperforms two-layer autoencoders for all $r > 0$. This picture is summarized in Figure 5.3: the blue curve represents the lower bound of Theorem 5 (for $r \leq 1$) and Proposition 5.4.2 (for $r > 1$), which is met by either running GD on the population risk (blue crosses) or SGD on samples taken from a isotropic Gaussian (green triangles) when $d = 100$;[3] this lower bound meets the performance of VAMP (red curve) if and only if $r \leq 1$; finally, the rate distortion function (orange curve) provides the best performance for all $r > 0$.

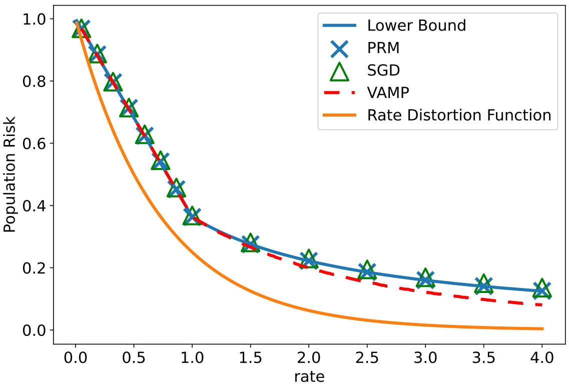


Figure 5.3: Performance comparison for the compression of an isotropic Gaussian source.

Universality of Gaussian predictions. Figure 5.3 (and also Figure 5.2 in Appendix C.7) show that gradient descent achieves the minimum of the population risk for the compression of Gaussian sources. Going beyond Gaussian inputs, to real-world datasets, Figure 5.1 (as well as those in Appendix C.7) show an excellent agreement between our predictions (using the empirical covariance of the data) and the performance of autoencoders trained on standard datasets (CIFAR-10, MNIST). As such, this agreement provides a clear indication of the universality of our predictions. In this regard, a flurry of recent research (see e.g. [HMRT22, HL22, LGC+21, GLR+22, DSL22, MS22, WZF24]) has proved that the Gaussian predictions actually hold in a much wider range of models. While none of the existing works exactly fits the setting considered in this paper, this gives yet another indication that our predictions should remain true more generally. The rigorous characterization of this universality is left for future work.

The choice of the activation function. The sign activation function constitutes an important special case of our analysis. However, our results hold for a broader class of activations. In particular, under the restriction that the rows of the encoder $\boldsymbol{B}$ lie on the unit sphere, all the results apply *for any odd activation*. The reason to fix the norm of the rows of the encoder is to prevent the network from entering the linear regime (e.g., by scaling $B \rightarrow \epsilon B$ and $A \rightarrow \frac{1}{\epsilon} A$). In fact, in the linear regime, perfect recovery can be achieved and this case has been well studied, see e.g. [BH89, KBGS19, GBLJ19]. We also note that, if the activation function is homogeneous, the restriction on the norm of the rows of $\boldsymbol{B}$ can be lifted, as the norm can be scaled out. Extending our analysis to activation functions that are not odd (e.g., ReLU) is an exciting avenue for future research. To achieve this goal, we expect that novel ideas will be needed, since our current approach relies on the fact that the Hermite expansion of the activation function (5.4) has only odd monomials.

[3] For further details on the experimental setup, see Appendix C.7.

CHAPTER 6

Autoencoders: Beyond Gaussian Data

In the previous chapter we discussed the compression of Gaussian data via shallow two-layer autoencoding. However, the Gaussian inputs lack any structure and may be viewed as the least "informative" distributionally. In this chapter, we ask the following question: to what degree does a shallow autoencoder capture the structure of the underlying data distribution? For the prototypical case of the 1-bit compression of sparse Gaussian data, we prove that gradient descent converges to a solution that completely disregards the sparse structure of the input. Namely, the performance of the algorithm is the same as if it was compressing a Gaussian source – with no sparsity. For general data distributions, we give evidence of a phase transition phenomenon in the shape of the gradient descent minimizer, as a function of the data sparsity: below the critical sparsity level, the minimizer is a rotation taken uniformly at random (just like in the compression of non-sparse data); above the critical sparsity, the minimizer is the identity (up to a permutation). Finally, by exploiting a connection with approximate message passing algorithms, we show how to improve upon Gaussian performance for the compression of sparse data: adding a denoising function to a shallow architecture already reduces the loss provably, and a suitable multi-layer decoder leads to a further improvement. We validate our findings on image datasets, such as CIFAR-10 and MNIST, along with particle physics dataset from [YM21b].

6.1 Motivation and Outlook

In this chapter, we continue the analysis of the shallow (GLM) encoding structure, however, for certain upcoming analysis the decoding map is no longer restricted to a linear transformation. In this view, it is beneficial to define a standalone notation for the encoding stage. Formally, we consider the encoding of $\boldsymbol{x} \in \mathbb{R}^d$ given by

$$\boldsymbol{z} = \sigma(\boldsymbol{B}\boldsymbol{x}), \quad \boldsymbol{B} \in \mathbb{R}^{n \times d}, \quad \boldsymbol{z} \in \mathbb{R}^n, \tag{6.1}$$

where the non-linear activation $\sigma(\cdot)$ is applied component-wise. The ratio $r = n/d$ is referred to as the compression rate. Recall that, for a shallow (two-layer) autoencoder, the decoding consists of a single linear transformation $\boldsymbol{A} \in \mathbb{R}^{d \times n}$:

$$\hat{\boldsymbol{x}}_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{z} = \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x}). \tag{6.2}$$

The optimal set of parameters $\boldsymbol{\Theta} = \{\boldsymbol{A}, \boldsymbol{B}\}$ minimizes the mean-squared error (MSE)

$$\mathcal{R}(\boldsymbol{\Theta}) := d^{-1}\mathbb{E}\left[\|\boldsymbol{x} - \hat{\boldsymbol{x}}_{\boldsymbol{\Theta}}(\boldsymbol{x})\|_2^2\right], \tag{6.3}$$

where the expectation is taken over the data distribution $\boldsymbol{x}$. The model described in (6.2) is a natural extension of *linear* autoencoders ($\sigma(x) = \alpha \cdot x$ for some $\alpha \neq 0$), which were thoroughly studied over the past years [KBGS19, GBLJ19, BLSG20]. In an effort to go beyond the linear setting, a number of recent works have considered the *non-linear* model (6.2). Specifically, [RG22, Ngu21] study the training dynamics under specific scaling regimes of the input dimension d and the number of neurons n, which lead to either vanishing or diverging compression rates. In the previous chapter we focus on the proportional regime in which d and n grow at the same speed, but our analysis relies heavily on Gaussian data assumptions. In contrast with Gaussian data that lacks any particular structure, real data often exhibits rich structural properties. For instance, images are inherently sparse, and this property has been exploited by various compression schemes such as jpeg. In this view, it is paramount to go beyond the analysis of unstructured Gaussian data and address the following fundamental questions:

> *Does gradient descent training of the two-layer autoencoder* (6.2) *capture the structure in the data? How does increasing the expressivity of the decoder impact the performance?*

To address these questions, we consider the compression of structured data via the non-linear autoencoder (6.1) with $\sigma \equiv \mathrm{sign}$ (1-bit compressed sensing, [BB08]) and show how the data structure is captured by the architecture of the decoder. Let us explain the choice of $\sigma \equiv \mathrm{sign}$. Apart from the connection to classical information and coding theory [CT06], its scale invariance prevents the model from entering the *linear regime* (see the details in Section 5.6 discussion on the choice of the activation function and in Section 5.3 setup paragraph). Thus, sign is a natural candidate to tackle the non-linear setting of interest in applications and, in fact, hard-thresholding activations are common in large-scale models [VDOV$^+$17].

Our main results can be summarized as follows:

- Theorem 9 proves that the *linear decoder* in (6.2) may be *unable to exploit the sparsity* in the data: when $\boldsymbol{x}$ has a Bernoulli-Gaussian (or "sparse Gaussian") distribution, both the gradient descent solution and the MSE coincide with those obtained for the compression of purely Gaussian data (with no sparsity).
- Going beyond Gaussian data, we give evidence of the emergence of a *phase transition* in the structure of the optimal matrices $\boldsymbol{A}, \boldsymbol{B}$ in (6.2), as the sparsity level $p \in (0, 1)$ varies: Proposition 6.4.1 locates the critical value of p such that the minimizer stops being a random rotation (as for purely Gaussian data), and it becomes the identity (up a permutation); numerical simulations for gradient descent corroborate this phenomenology and display a "staircase" behavior of the loss function.
- While for the compression of sparse Gaussian data the linear decoder in (6.2) does not capture the sparsity, we show in Section 6.5 that increasing the expressivity of the decoder improves upon Gaussian performance. First, we post-process the output of (6.2), i.e., we consider

  $$\hat{\boldsymbol{x}}_{\boldsymbol{\Theta}}(\boldsymbol{x}) = f(\boldsymbol{A}\boldsymbol{z}) = f(\boldsymbol{A}\mathrm{sign}(\boldsymbol{B}\boldsymbol{x})), \tag{6.4}$$

  where f is applied component-wise, and we prove that a suitable f leads to a smaller MSE. In other words, adding a *nonlinearity* to the linear decoder in (6.2) provably helps.

Finally, we further improve the performance by increasing the *depth* and using a multi-layer decoder. Our analysis leverages a connection between multi-layer autoencoders and the iterates of the RI-GAMP algorithm proposed by [VKM22], which may be of independent interest.

Experiments on syntethic data confirm our findings, and similar phenomena are displayed when running gradient descent to compress CIFAR-10/MNIST images and particle physics data [YM21b]. Taken together, our results show that, for the compression of structured data, a more expressive decoding architecture provably improves performance. This is in sharp contrast with the compression of unstructured, Gaussian data where, as discussed in Section 6 of [SKHM23], multiple decoding layers do not help.

6.2 Related Work

Theoretical results for autoencoders. For the detailed related work regarding the theory of autoencoding we refer to the previous discussion in Section 5.2. The only addition corresponds to the results reported in the previous chapter. To recap briefly, we consider the compression of Gaussian data with a two-layer autoencoder when the compression rate r is fixed and show that gradient descent methods achieve a minimizer of the MSE. We provide a tight lower bound on mean squared error achieved by the model and characterize the corresponding SGD training dynamics in the case of isotropic data for $r \leq 1$. We also indicate a universality of Gaussian predictions on natural data that is closely related to inability of two-layer autoencoders to capture of certain input structure which we outline for sparse Gaussian and sparse Rademacher signals with a certain sparsity thresholds later on.

Incremental learning and staircases in the training dynamics. Phenomena similar to the staircase behavior of the loss function that we exhibit in Figure 6.3 have drawn significant attention. For parity learning, the line of works [ABAB$^+$21, AAM22, AAM23] shows that parities are recovered in a sequential fashion with increasing complexity. A similar behaviour is observed in transformers with diagonal weight matrices at small initialization [ABBA$^+$23]: gradient descent progressively learns a solution of increasing rank. For a single index model, [BMZ23] show a separation of time-scales at which the training dynamics follows an alternating pattern of plateaus and rapid decreases in the loss. Evidence of incremental learning in deep linear networks is provided by [Ber23, PF23, SKZ$^+$23, JGȘ$^+$21, MKAA21]. The recent work by [SBGG23] shows that the cumulants of the data distribution are learnt sequentially, revealing a sample complexity gap between neural networks and random features.

Approximate Message Passing (AMP). AMP refers to a family of iterative algorithms developed for a variety of statistical inference problems [FVR$^+$22]. Such problems include the recovery of a signal $\boldsymbol{x}$ from observations $\boldsymbol{z}$ of the form in (6.1), namely, a Generalized Linear Model [MN89], when the encoder matrix $\boldsymbol{B}$ is Gaussian [Ran11, MV22] or rotationally-invariant [RSF19, SRF16, MP17, Tak19]. Of particular interest for our work is the RI-GAMP algorithm by [VKM22]. In fact, RI-GAMP enjoys a computational graph structure that can be mapped to a suitable neural network, and it approaches the information-theoretically optimal MSE. The optimal MSE was computed via the replica method by [TUK06, TCVS13], and these predictions were rigorously confirmed for the high-temperature regime by [LFSW23]. For the complimentary details also see Section 2.3 of Chapter 2.

6.3 Preliminaries

Notation. We use plain symbols a, b for scalars, bold symbols $\boldsymbol{a}, \boldsymbol{b}$ for vectors, and capitalized bold symbols $\boldsymbol{A}, \boldsymbol{B}$ for matrices. Given a vector $\boldsymbol{a}$, its ℓ_2-norm is $\|\boldsymbol{a}\|_2$. Given a matrix $\boldsymbol{A}$, its operator norm is $\|\boldsymbol{A}\|_{op}$. We denote a unidimensional Gaussian distribution with mean μ and variance σ^2 by $\mathcal{N}(\mu, \sigma^2)$. We use the shorthand $\hat{\boldsymbol{x}}$ for $\hat{\boldsymbol{x}}_{\boldsymbol{\Theta}}$. Unless specified otherwise, function are applied *component-wise* to vector/matrix-valued inputs. We denote by $C, c > 0$ universal constants, which are independent of n, d.

Data distribution and MSE. For $p \in (0, 1]$, a sparse Gaussian distribution $\mathrm{SG}_1(p)$ is equal to $\mathcal{N}(0, 1/p)$ with probability p and is 0 otherwise. The scaling of the variance of the Gaussian component ensures a unit second moment for all p. We use the notation $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ to denote a vector with i.i.d. components distributed according to $\mathrm{SG}_1(p)$. Decreasing p makes $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ more sparse: for $p = 1$ one recovers the isotropic Gaussian, i.e., $\mathrm{SG}_d(1) \equiv \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, while $p = 0$ implies that $\boldsymbol{x} = \boldsymbol{0}$.

In Chapter 5, we consider Gaussian data $\boldsymbol{x} \sim \mathrm{SG}_d(1)$ and the two-layer autoencoder with linear decoder in (6.2). Our analysis unveils that, for a compression rate $r \leq 1$, the MSE obtained by minimizing (6.3) over $\boldsymbol{\Theta} = \{\boldsymbol{A}, \boldsymbol{B}\}$ is given by

$$\mathcal{R}_{\mathrm{Gauss}} := 1 - \frac{2}{\pi} \cdot r. \tag{6.5}$$

The set of minimizers $(\boldsymbol{A}, \boldsymbol{B})$ has a weight-tied orthogonal structure, i.e., $\boldsymbol{B}\boldsymbol{B}^\top = \boldsymbol{I}$ and $\boldsymbol{A} \propto \boldsymbol{B}^\top$, and gradient-based optimization schemes reach a global minimum.

6.4 Limitations of a Linear Decoding Layer

Our main technical result is that a two-layer autoencoder with a single linear decoding layer does not capture the sparse structure of the data. Specifically, we consider the autoencoder in (6.2) with Gaussian data $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ trained via gradient descent. We show that, when n, d are both large (holding the compression rate $r = n/d$ fixed), the trajectory of the algorithm is the same as that obtained from the compression of non-sparse data, i.e., $\boldsymbol{x} \sim \mathrm{SG}_d(1) \equiv \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. As a consequence, the minimizer has a weight-tied orthogonal structure ($\boldsymbol{B}\boldsymbol{B}^\top = \boldsymbol{I}$, $\boldsymbol{A} \propto \boldsymbol{B}^\top$), and the MSE at convergence is given by $\mathcal{R}_{\mathrm{Gauss}}$ as defined in (6.5).

We now go into the details. Since the optimization objective is convex in $\boldsymbol{A}$, we consider the following alternating minimization version of Riemannian gradient descent:

$$\begin{aligned} \boldsymbol{A}(t+1) &= \arg\min_{\boldsymbol{A}} \mathcal{R}(\boldsymbol{A}, \boldsymbol{B}(t)), \\ \boldsymbol{B}(t+1) &:= \mathrm{proj}\left(\boldsymbol{B}(t) - \eta\left(\nabla_{\boldsymbol{B}(t)} + \boldsymbol{G}(t)\right)\right). \end{aligned} \tag{6.6}$$

In fact, due to the convexity in $\boldsymbol{A}$ of the MSE $\mathcal{R}(\cdot, \cdot)$ in (6.3), we can compute in closed form $\arg\min_{\boldsymbol{A}} \mathcal{R}(\boldsymbol{A}, \boldsymbol{B}(t))$. Here, Riemannian refers to the space of matrices with unit-norm rows, $\nabla_{\boldsymbol{B}(t)}$ is a shorthand for the gradient $\nabla_{\boldsymbol{B}(t)}\mathcal{R}(\boldsymbol{A}(t), \boldsymbol{B}(t))$, and proj normalizes the rows of a matrix to have unit norm. The projection step (and, hence, the Riemannian nature of the optimization) is due to the scale-invariance of sign, and it ensures numerical stability. The term $\boldsymbol{G}(t)$ corresponds to Gaussian noise of arbitrarily small variance, which acts as a (probabilistic) smoothing for the discontinuity of sign at 0 and, therefore, implies that the gradient is well-defined along the trajectory of the algorithm. (Note that $\boldsymbol{G}(t)$ is not needed in experiments, as we use a straight-through estimator, see Appendix D.2.1).

Theorem 9 (Gradient descent does not capture the sparsity). *Consider the gradient descent algorithm in* (6.6) *with* $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ *and* $(\boldsymbol{G}(t))_{i,j} \sim \mathcal{N}(0, \sigma^2)$, *where* $d^{-\gamma_g} \leq \sigma \leq C/d$ *for some fixed* $1 < \gamma_g < \infty$. *Initialize the algorithm with* $\boldsymbol{B}(0)$ *equal to a row-normalized Gaussian, i.e.,* $\boldsymbol{B}'_{i,j}(0) \sim \mathcal{N}(0, 1/d)$, $\boldsymbol{B}(0) = \mathrm{proj}(\boldsymbol{B}'(0))$, *and let* $\boldsymbol{B}(0) = \boldsymbol{U}\boldsymbol{S}(0)\boldsymbol{V}^\top$ *be its SVD. Let the step size* η *be* $\Theta(1/\sqrt{d})$. *Then, for any fixed* $r < 1$ *and* $T_{\max} \in (0, \infty)$, *with probability at least* $1 - Cd^{-3/2}$, *the following holds for all* $t \leq T_{\max}/\eta$

$$\begin{aligned} &\boldsymbol{B}(t) = \boldsymbol{U}\boldsymbol{S}(t)\boldsymbol{V}^\top + \boldsymbol{R}(t), \\ &\left\|\boldsymbol{S}(t)\boldsymbol{S}(t)^\top - \boldsymbol{I}\right\|_{op} \leq C\exp\left(-c\eta t\right), \\ &\lim_{d\to\infty} \sup_{t\in[0,T_{\max}/\eta]} \left\|\boldsymbol{R}(t)\right\|_{op} = 0, \end{aligned} \tag{6.7}$$

where C, c *are universal constants depending only on* p, r *and* $T_{\max}$. *Moreover, we have that, almost surely,*

$$\lim_{t\to\infty}\lim_{d\to\infty} \mathcal{R}(\boldsymbol{A}(t), \boldsymbol{B}(t)) = \mathcal{R}_{\mathrm{Gauss}}, \tag{6.8}$$

$$\lim_{d\to\infty} \sup_{t\in[0,T_{\max max}/\eta]} \left\|\boldsymbol{B}(t) - \boldsymbol{B}_{\mathrm{Gauss}}(t)\right\|_{op} = 0, \tag{6.9}$$

where $\mathcal{R}_{\mathrm{Gauss}}$ *is defined in* (6.5) *and* $\boldsymbol{B}_{\mathrm{Gauss}}(t)$ *is obtained by running* (6.6) *with* $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$.

In words, (6.7) gives a precise characterization of the gradient descent trajectory: throughout the dynamics, the eigenbasis of $\boldsymbol{B}(t)$ does not change significantly (i.e., it remains close to that of $\boldsymbol{B}(0)$) and, as t grows, all the singular values of $\boldsymbol{B}(t)$ approach 1. As a consequence, (6.8) gives that, at convergence, the MSE achieved by (6.6) with $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ approaches $\mathcal{R}_{\mathrm{Gauss}}$, which corresponds to the compression of standard Gaussian data $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. In fact, a stronger result holds: (6.9) gives that the whole trajectory of (6.6) for $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ is the same as that obtained for $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$.

The fact that the autoencoder model in (6.2) is not able to exploit the signal structure is quite surprising, especially since information-theoretically a sparse Gaussian source is more suitable for compression than its non-sparse counterpart. Namely, for sparse Gaussian data, one can compute *rate-distortion function*, which is the information-theoretically optimal MSE that can be achieved for a given compression rate. This is done via the Blahut-Arimoto algorithm [Bla72, Ari72] in Figure 6.1. We observe that as sparsity increase, the optimal MSE decreases, so the data is easier to compress.

Beyond Gaussian data: Phase transitions, staircases in the learning dynamics, and image data. For general distributions of the data $\boldsymbol{x}$, we empirically observe that the minimizers of the model in (6.2) found by stochastic gradient descent (SGD) either *(i)* coincide with those obtained for standard Gaussian data, or *(ii)* are equivalent to (suitably sub-sampled) permutations of the identity. Up to a permutation of the neurons, these two candidates can be expressed as:

$$\begin{aligned} &\hat{\boldsymbol{x}}_{\mathrm{Haar}}(\boldsymbol{x}) = \alpha_{\mathrm{Haar}} \cdot \boldsymbol{B}^\top \mathrm{sign}(\boldsymbol{B}\boldsymbol{x}), \\ &\hat{\boldsymbol{x}}_{\mathrm{Id}}(\boldsymbol{x}) = \alpha_{\mathrm{Id}} \cdot \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0}_{(d-n)\times n} \end{bmatrix} \mathrm{sign}([\boldsymbol{I}_n, \boldsymbol{0}_{n\times(d-n)}]\boldsymbol{x}), \end{aligned} \tag{6.10}$$

where $\boldsymbol{B}$ is obtained by subsampling a Haar matrix (i.e., a matrix taken uniformly from the group of rotations), $\boldsymbol{0}_{(d-n)\times n}$ is a $(d-n) \times n$ matrix of zeros, and $(\alpha_{\mathrm{Haar}}, \alpha_{\mathrm{Id}})$ are scalar coefficients. The losses of these two candidates can be expressed in a closed form as derived below.

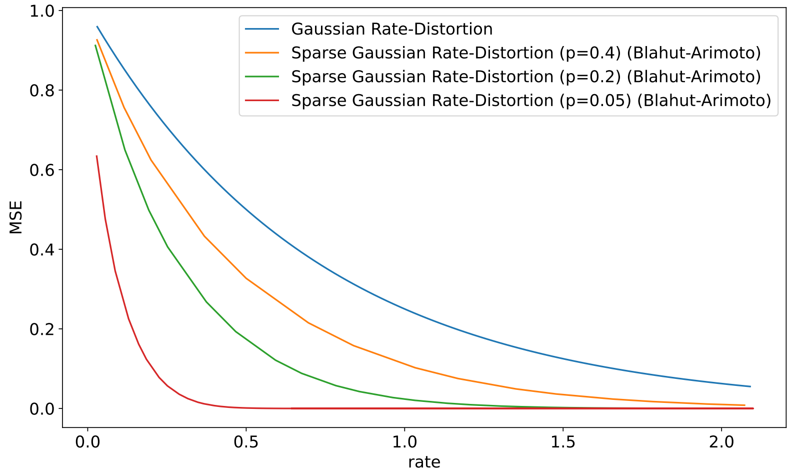


Figure 6.1: Numerical computation of the rate-distortion function for a sparse Gaussian source via the Blahut-Arimoto algorithm. We plot the optimal MSE against the rate r for different values of sparsity p.

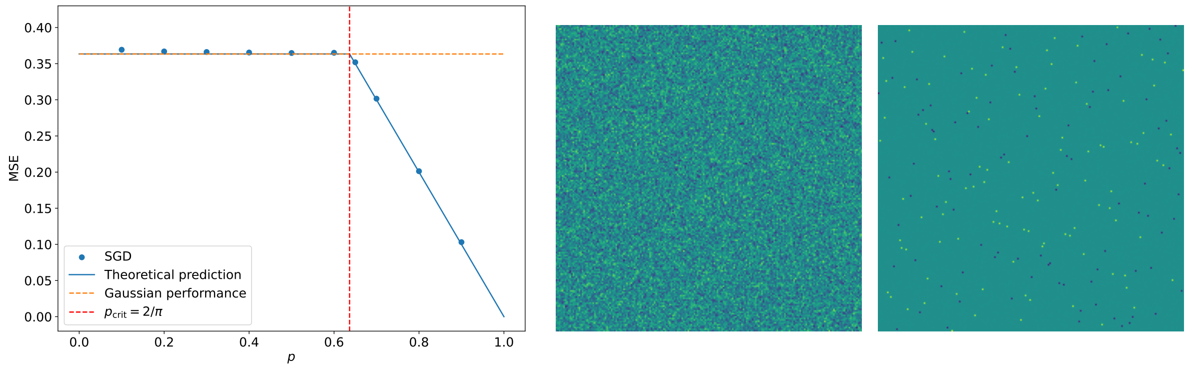


Figure 6.2: Compression of sparse Rademacher data via the two-layer autoencoder in (6.2). We set $d = 200$ and $r = 1$. *Left.* MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line): for $p < p_{\mathrm{crit}}$, the loss is equal to the value obtained for Gaussian data, i.e., $\mathcal{R}_{\mathrm{Gauss}} = 1 - 2r/\pi$; for $p \geq p_{\mathrm{crit}}$, the loss is smaller, and it is equal to $1 - r \cdot (\mathbb{E}|x_1|)^2 = 1 - r \cdot p$. *Center.* Encoder matrix $\boldsymbol{B}$ at convergence of SGD when $p = 0.3 < p_{\mathrm{crit}}$: the matrix is a random rotation. *Right.* Encoder matrix $\boldsymbol{B}$ at convergence of SGD when $p = 0.7 \geq p_{\mathrm{crit}}$: the negative sign in part of the entries of $\boldsymbol{B}$ is cancelled by the corresponding sign in the entries of $\boldsymbol{A}$; hence, $\boldsymbol{B}$ is equivalent to a permutation of the identity.

Proposition 6.4.1 (Candidate comparison)**.** *Let $r \leq 1$ and let $\boldsymbol{x}$ have i.i.d. components with zero mean and unit variance. Then, we have that, almost surely, the MSE of $\hat{\boldsymbol{x}}_{\mathrm{Haar}}(\cdot)$ coincides with the Gaussian performance $\mathcal{R}_{\mathrm{Gauss}}$ in* (6.5)*, i.e.,*

$$\min_{\alpha_{\mathrm{Haar}} \in \mathbb{R}} \lim_{d \to \infty} \frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \left[\|\hat{\boldsymbol{x}}_{\mathrm{Haar}}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \right] = 1 - \frac{2}{\pi} \cdot r \,. \tag{6.11}$$

Furthermore, we have that, for any d,

$$\min_{\alpha_{\mathrm{Id}} \in \mathbb{R}} \frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \left[\|\hat{\boldsymbol{x}}_{\mathrm{Id}}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \right] = 1 - r \cdot (\mathbb{E}|x_1|)^2 \,, \tag{6.12}$$

where x_1 is the first component of $\boldsymbol{x}$. This implies that $\hat{\boldsymbol{x}}_{\mathrm{Id}}(\cdot)$ is superior to $\hat{\boldsymbol{x}}_{\mathrm{Haar}}(\cdot)$ in terms of MSE whenever

$$\mathbb{E}|x_1| > \sqrt{2/\pi} = \mathbb{E}_{g \sim \mathcal{N}(0,1)} |g|. \tag{6.13}$$

The MSE of $\hat{\boldsymbol{x}}_{\rm Id}(\cdot)$ in (6.12) is obtained via a direct calculation. To evaluate the MSE of $\hat{\boldsymbol{x}}_{\rm Haar}(\cdot)$ in (6.11), we relate this estimator to the first iterate of the RI-GAMP algorithm proposed by [VKM22]. Then, the high dimensional limit of $\|\alpha_{\rm Haar} \cdot \boldsymbol{B}^\top {\rm sign}(\boldsymbol{B}\boldsymbol{x}) - \boldsymbol{x}\|_2^2$ follows from the state evolution analysis of RI-GAMP. A similar strategy will be used also in Section 6.5 to analyze different decoding architectures. The complete proof is in Appendix D.1.1.

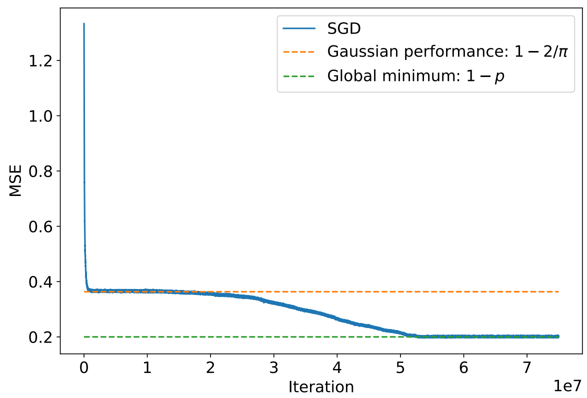


Figure 6.3: Compression of sparse Rademacher data via the two-layer autoencoder in (6.2). We set $d = 200$, $r = 1$ and $p = 0.8$. The MSE is plotted as a function of the number of iterations and, as $p > p_{\rm crit}$, it displays a staircase behavior.

As mentioned above, our numerical results lead us to conjecture that SGD recovers either of the candidates in (6.10), depending on which achieves a smaller loss. Specifically, if condition (6.13) is met, the SGD predictor converges to $\hat{\boldsymbol{x}}_{\rm Id}(\cdot)$ and improves upon the Gaussian loss $\mathcal{R}_{\rm Gauss}$; otherwise, it converges to $\hat{\boldsymbol{x}}_{\rm Haar}(\cdot)$ and its MSE is equal to $\mathcal{R}_{\rm Gauss}$.

For sparse Gaussian data, condition (6.13) is never satisfied, as $\mathbb{E}_{x_1\sim {\rm SG}_1(p)}|x_1| = \sqrt{2p/\pi} \leq \sqrt{2/\pi}$. In fact, as proved in Theorem 9, the SGD solution approaches $\hat{\boldsymbol{x}}_{\rm Haar}(\cdot)$ and its MSE matches $\mathcal{R}_{\rm Gauss}$.

For sparse Rademacher data[1], condition (6.13) reduces to $p > p_{\rm crit} := 2/\pi \approx 0.64$, and Figure 6.2 shows a *phase transition* in the structure of the minimizers found by SGD:

- For $p < p_{\rm crit}$, SGD converges to a solution s.t. $\boldsymbol{B}$ is a uniform rotation (central heatmap) and the MSE is close to $\mathcal{R}_{\rm Gauss} = 1 - 2r/\pi$, see (6.11).
- For $p > p_{\rm crit}$, SGD converges to a solution s.t. $\boldsymbol{B}$ is equivalent to a permutation of the identity (right heatmap) and the MSE is close to $1 - r \cdot (\mathbb{E}|x_1|)^2 = 1 - r \cdot p$, see (6.12). In both cases, $\boldsymbol{A} \propto \boldsymbol{B}^\top$.

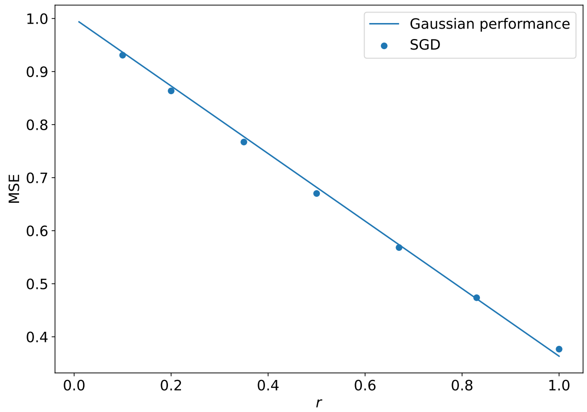


Figure 6.4: Compression of whitened physics particle data [YM21b] via the two-layer autoencoder in (6.2). The SGD loss at convergence (dots) matches the solid line, which corresponds to the prediction in (6.5) for the compression of standard Gaussian data (with no sparsity).

If there is an improvement upon $\mathcal{R}_{\rm Gauss}$ (i.e., $p > p_{\rm crit}$), the SGD dynamics exhibits a *staircase* behavior. This phenomenon is displayed in Figure 6.3, which plots the error as a function of the number of SGD iterations for $p = 0.8 > p_{\rm crit}$: first, the MSE rapidly converges to $\mathcal{R}_{\rm Gauss}$; then, there is a plateau; finally, the global minimum $1 - r \cdot p$ is reached. We also remark that, as p approaches $p_{\rm crit}$, the time needed by SGD to escape the plateau increases. A possible explanation is that, as p decreases, the noise due to masking increases, which increases the variance of the gradient. This makes it harder for $\boldsymbol{B}$ to find a direction towards a permutation of the identity (i.e., the global minimum). Additional evidence of both the

[1] Each i.i.d. component is equal to 0 w.p. $1 - p$ and to $\pm 1/\sqrt{p}$ w.p. $p/2$, which ensures a unit second moment for all $p \in [0, 1]$.

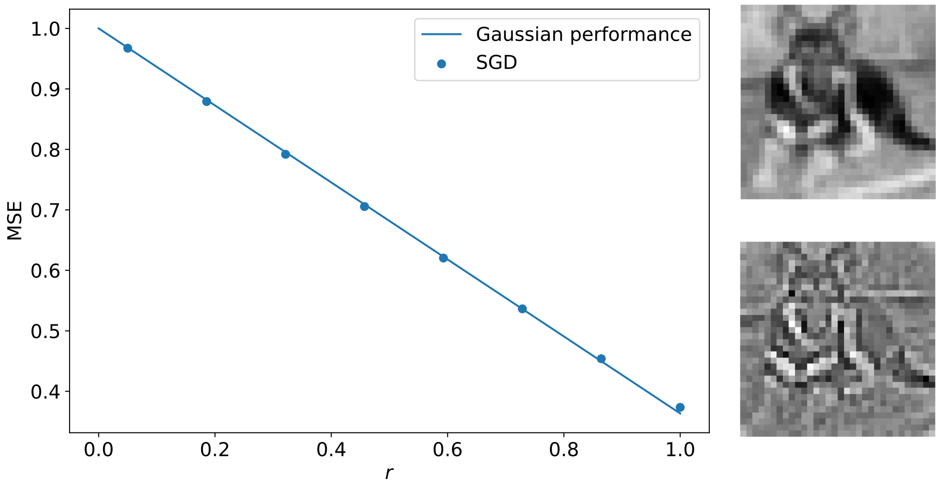


Figure 6.5: Compression of masked and whitened CIFAR-10 images of the class “dog” via the two-layer autoencoder in (6.2). First, the data is whitened so that it has identity covariance (as in the setting of Theorem 9). Then, the data is masked by setting each pixel independently to 0 with probability $p = 0.7$. An example of an original image is on the top right, and the corresponding masked and whitened image is on the bottom right. The SGD loss at convergence (dots) matches the solid line, which corresponds to the prediction in (6.5) for the compression of standard Gaussian data (with no sparsity).

phase transition and the staircase behavior of SGD is in Appendix D.2.2, where Figure D.1 considers Rademacher data, Figures D.2-D.3 data coming from a sparse mixture of Gaussians, Figure D.4 data sampled from a sparse mixture of Beta distributions, Figures D.5 and D.6 data distributed according to a (non-sparse) mixture of Gaussians with varying aspect ratio, and Figure D.7 sparse Laplace data.

The proof strategy of Theorem 9 could be useful to track SGD until it reaches the plateau. However, characterizing the time-scale needed to escape the plateau likely requires new tools, and it provides an exciting research direction.

Finally, Figure 6.5 shows that our theory predicts well the behavior of the compression of *CIFAR-10 images* via the two-layer autoencoder in (6.2). We let x_1 be the empirical distribution of the image pixels after whitening and masking[2], and we verify that condition (6.13) does not hold. Then, as expected, the autoencoder is unable to capture the structure coming from masking part of the pixels, and the loss at the end of SGD training equals $\mathcal{R}_{\text{Gauss}}$. Similar results hold for MNIST, see Figure D.8 in Appendix D.2.3, and particle physics data [YM21b], see Figure 6.4.

6.5 Provable Benefit of Nonlinearities and Depth

In this section, we prove that more expressive decoders than the linear one in (6.2) capture the sparsity of the data and, therefore, improve upon the Gaussian loss $\mathcal{R}_{\text{Gauss}}$.

[2] The whitening makes the data have isotropic covariance, as required by our theory; the masking makes the data sparse.

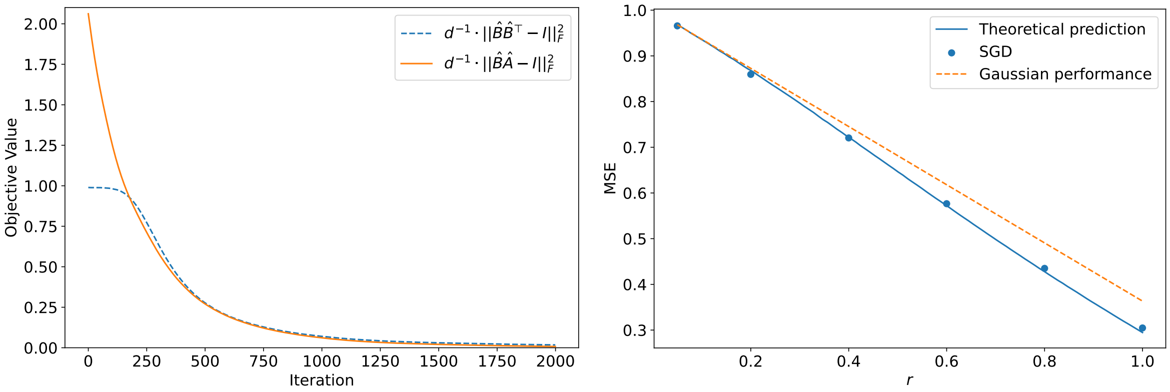


Figure 6.6: Compression of sparse Gaussian data via the autoencoder in (6.4), where f has the form in (6.15) and its parameters $(\alpha_1, \alpha_2, \alpha_3)$ are optimized via SGD. We set $d = 100$ and $p = 0.4$. *Left.* Distance between $\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top$, $\hat{\boldsymbol{B}}\hat{\boldsymbol{A}}$ and the identity, as a function of the number of iterations, where $\hat{\boldsymbol{B}}$, $\hat{\boldsymbol{A}}$ denote the row-normalized versions of $\boldsymbol{B}$, $\boldsymbol{A}$. $\|\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top - \boldsymbol{I}\|_F$ and $\|\hat{\boldsymbol{B}}\hat{\boldsymbol{A}} - \boldsymbol{I}\|_F$ decrease and tend to 0, meaning that (up to a rescaling of the rows) $\boldsymbol{B}\boldsymbol{A}$ and $\boldsymbol{B}\boldsymbol{B}^\top$ approach the identity. Here, we take $r = 1$. *Right.* MSE achieved by SGD at convergence, as a function of the compression rate r. The empirical values (dots) match the characterization of Proposition 6.5.1 for $f = f^*$ in (6.18) (blue line), and they outperform the MSE (6.5) obtained by compressing standard Gaussian data (orange dashed line).

6.5.1 Provable Benefit of Nonlinearities

First, we apply a nonlinearity at the output of the linear decoding layer, as in (6.4). The ResNet-like denoising architecture analyzed in [CZ23] suggests a suitable choice of the non-linearity. The corresponding denoising network has the following form:

$$\boldsymbol{x}\cdot\alpha + \boldsymbol{\Theta}_1\cdot\tanh(\boldsymbol{\Theta}_2\cdot\boldsymbol{x}). \tag{6.14}$$

To map (6.14) to a scalar denoising function, we fix $\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2 \propto \boldsymbol{I}$ (or a row-subsampled version of an identity matrix for $r < 1$). Specifically, we take

$$f(x) = \alpha_1 x + \alpha_2 \tanh(\alpha_3 x), \tag{6.15}$$

and run SGD on the weight matrices $(\boldsymbol{A}, \boldsymbol{B})$ and on the trainable parameters $(\alpha_1, \alpha_2, \alpha_3)$ in f. Figure 6.6 shows that, at convergence, the minimizers have the same weight-tied orthogonal structure as obtained for Gaussian data ($\boldsymbol{B}\boldsymbol{B}^\top = \boldsymbol{I}$, $\boldsymbol{A} \propto \boldsymbol{B}^\top$), see the left plot. However, in sharp contrast with Gaussian data, the loss is *smaller* than $\mathcal{R}_{\text{Gauss}}$, see the blue dots on the right plot and compare them with the orange dashed curve. This empirical evidence motivates us to analyze the performance of autoencoders of the form (6.4), where $\boldsymbol{B}$ is obtained by subsampling a Haar matrix of appropriate dimensions and $\boldsymbol{A} = \boldsymbol{B}^\top$.

Proposition 6.5.1 (MSE characterization)**.** *Let $r \le 1$ and $\boldsymbol{x}$ have i.i.d. components with zero mean and unit variance. Consider the autoencoder $\hat{\boldsymbol{x}}(\boldsymbol{x})$ in* (6.4)*, where $\boldsymbol{B}$ is obtained by subsampling a Haar matrix, $\boldsymbol{A} = \boldsymbol{B}^\top$, and f is a Lipschitz function. Then, we have that, almost surely,*

$$\lim_{d\to\infty}\frac{1}{d}\cdot\mathbb{E}_{\boldsymbol{x}}\|\boldsymbol{x} - \hat{\boldsymbol{x}}(\boldsymbol{x})\|_2^2 = \mathbb{E}_{x_1,g}|x_1 - f(\mu x_1 + \sigma g)|_2^2, \tag{6.16}$$

where x_1 is the first entry of $\boldsymbol{x}$, $g \sim \mathcal{N}(0,1)$ and independent of x_1, and the parameters (μ, σ) are given by

$$\mu = r\sqrt{\frac{2}{\pi}}, \quad \sigma^2 = r\left(1 - r\cdot\frac{2}{\pi}\right) > 0. \tag{6.17}$$

Proposition 6.5.1 is a generalization of Proposition 6.4.1, which corresponds to taking a linear f. The idea is to relate $f(\boldsymbol{B}^\top \mathrm{sign}(\boldsymbol{B}\boldsymbol{x}))$ to the first iterate of a suitable RI-GAMP algorithm, so that the characterization in (6.16) follows from state evolution. The details are in Appendix D.1.2.

Armed with Proposition 6.5.1, one can readily establish the function f that minimizes the MSE for large d. This in fact corresponds to the f that minimizes the RHS of (6.16), i.e.,

$$f^*(y) = \mathbb{E}[x_1 | \mu x_1 + \sigma g = y], \tag{6.18}$$

as long as the latter is Lipschitz (so that Proposition 6.5.1 can be applied). Sufficient conditions for f^* to be Lipschitz are that either *(i)* x_1 has a log-concave density, or *(ii)* there exist independent random variables u_0, v_0 s.t. u_0 is Gaussian, v_0 is compactly supported and x_1 is equal in distribution to $u_0 + v_0$, see Lemma 3.8 of [FVR$^+$22]. The expression of f^* for distributions of x_1 considered in the experiments (sparse Gaussian, Laplace, and Rademacher) is derived in Appendix D.1.4.

The blue curve in the right plot of Figure 6.6 evaluates the RHS of (6.16) for the optimal $f = f^*$, when $x_1 \sim \mathrm{SG}_1(p)$. Two observations are in order:

1. The blue curve matches the blue dots, obtained by optimizing via SGD the matrices $\boldsymbol{A}, \boldsymbol{B}$ and f in the parametric form (6.15). This means that the SGD performance is accurately tracked by plugging the optimal function (6.18) into the prediction of Proposition 6.5.1.

2. The blue curve improves upon the Gaussian loss $\mathcal{R}_{\mathrm{Gauss}}$ (orange dashed line). This means that, while the two-layer autoencoder in (6.2) is stuck at the MSE in orange (as proved by Theorem 9), by incorporating a nonlinearity, the autoencoder in (6.4) does better. In fact, as shown in Figure D.10 in Appendix D.2.5, the MSE achieved by the autoencoder in (6.4) with the optimal choice of f (namely, the RHS of (6.16) with $f = f^*$) is strictly lower than $\mathcal{R}_{\mathrm{Gauss}}$ for any $p \in (0, 1)$.

Beyond Gaussian data: Phase transitions, staircases in the learning dynamics, and image data. For general data $\boldsymbol{x}$ with i.i.d. zero-mean unit-variance components, the autoencoder in (6.4) displays a behavior similar to that described in Section 6.4 for the autoencoder in (6.2): the SGD minimizers of the weight matrices $\boldsymbol{A}, \boldsymbol{B}$ either exhibit a weight-tied orthogonal structure ($\boldsymbol{B}\boldsymbol{B}^\top = \boldsymbol{I}$, $\boldsymbol{A} \propto \boldsymbol{B}^\top$), or come from permutations of the identity. This leads to a *phase transition* in the structure of the minimizer (and in the MSE expression), as the sparsity p varies. To quantify the critical value of p at which the minimizer changes, one can compare the MSE when $\boldsymbol{B}$ is subsampled *(i)* from a Haar matrix, and *(ii)* from the identity. The former is readily obtained from Proposition 6.5.1 where f is given by (6.18), and the latter is given by the result below, which is proved in Appendix D.1.3.

Proposition 6.5.2. *Let $\boldsymbol{x}$ have i.i.d. components with zero mean, unit variance and a symmetric distribution (i.e., the law of x_1 is the same as that of $-x_1$). Define $\hat{\boldsymbol{x}}_{\mathrm{Id}}(\boldsymbol{x})$ as in* (6.10)*, and fix $r \le 1$. Then, we have that, for any d,*

$$\min_f \frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \left[\| f(\hat{\boldsymbol{x}}_{\mathrm{Id}}(\boldsymbol{x})) - \boldsymbol{x} \|_2^2 \right] = 1 - r \cdot (\mathbb{E}|x_1|)^2. \tag{6.19}$$

Figure 6.7 displays the phase transition for the compression of sparse Rademacher data:

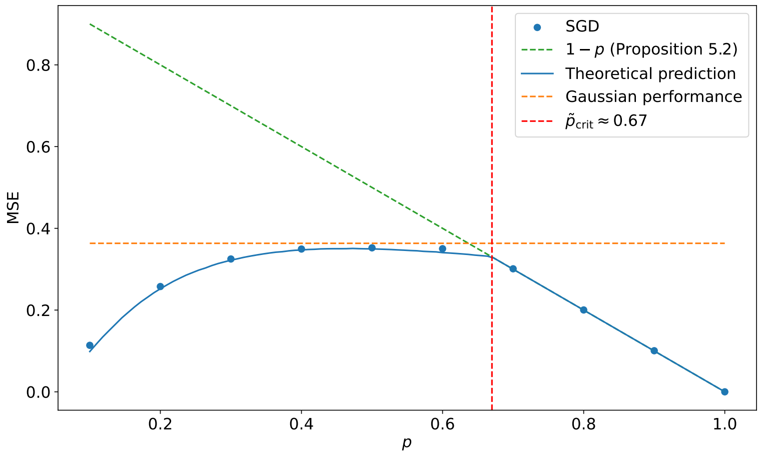


Figure 6.7: Compression of sparse Rademacher data via the autoencoder in (6.4). We set $d = 200$ and $r = 1$. The MSE achieved by SGD at convergence is plotted as a function of the sparsity level p. The empirical values (blue dots) match our theoretical prediction (blue line). For $p < \tilde{p}_{\text{crit}}$, the MSE is given by Proposition 6.5.1 for $\boldsymbol{B}$ sampled from the Haar distribution; for $p \geq \tilde{p}_{\text{crit}}$, the MSE is given by Proposition 6.5.2 for $\boldsymbol{B}$ equal to the identity.

- For $p < \tilde{p}_{\text{crit}} \approx 0.67$, SGD converges to a solution with MSE given by the RHS of (6.16) with $f = f^*$. Furthermore, $\boldsymbol{B}$ is a uniform rotation (see the central heatmap in Figure D.11 of Appendix D.2.6).
- For $p > \tilde{p}_{\text{crit}}$, SGD converges to a solution with MSE given by the RHS of (6.19). Furthermore, $\boldsymbol{B}$ is equivalent to a permutation of the identity (see the right heatmap in Figure D.11 of Appendix D.2.6).

By comparing the blue dots/curve with the orange dashed line in Figure 6.7, we also conclude that, for all p, the MSE of the autoencoder in (6.4) improves upon the Gaussian performance $\mathcal{R}_{\text{Gauss}}$. This is in contrast with the behavior of the autoencoder in (6.2) which remains stuck at $\mathcal{R}_{\text{Gauss}}$ for $p < 2/\pi$ (see Figure 6.2), and it demonstrates the benefit of adding the nonlinearity f.

For $p > \tilde{p}_{\text{crit}}$, the learning dynamics exhibits again a *staircase* behavior in which the MSE first gets stuck at the value given by the RHS of (6.16) with $f = f^*$, and then reaches the optimal value of $1 - r \cdot (\mathbb{E}|x_1|)^2$. This is reported for $p = 0.9 > \tilde{p}_{\text{crit}} \approx 0.67$ in Figure D.13 of Appendix D.2.6.

Additional numerical simulations to demonstrate both phase transition and staircase behaviour for the autoencoder in 6.4 are presented in Appendix D.2.6. Namely, Figure D.14 corresponds to data coming from a sparse mixture of Beta distributions, Figure D.15 considers sparse Gaussian mixture data, Figure D.16 considers the setting of (non-sparse) Gaussian mixture with varying aspect ratio, and Figure D.17 illustrates the results achieved on sparse Laplace data.

Finally, Figure 6.8 shows that the key features we unveiled for the autoencoder in (6.4) are still present when compressing *sparse CIFAR-10 data*. The empirical distribution of the image pixels after whitening is well approximated by a Laplace random variable (see Figure D.9 in Appendix D.2.4), thus we denote by x_1 the corresponding sparse Laplace distribution (see (D.7) in Appendix D.1.4 for a formal definition). The encoder matrix $\boldsymbol{B}$ is obtained by subsampling

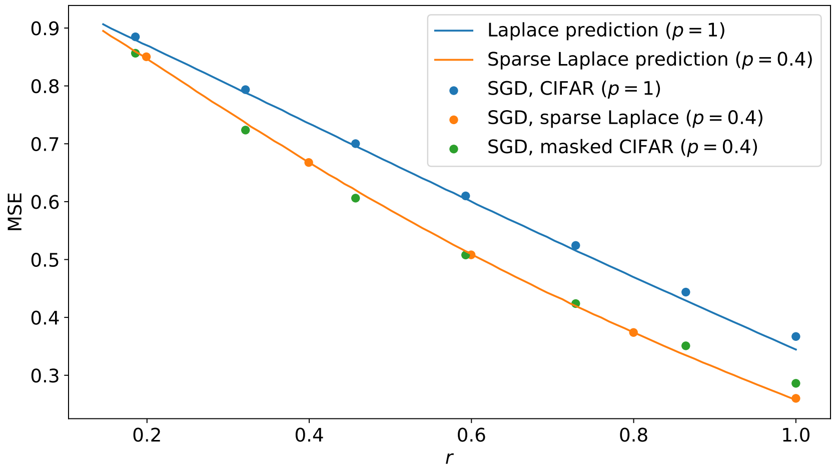


Figure 6.8: Compression of masked and whitened CIFAR-10 images of the class "dog" via the autoencoder in (6.4). We plot the MSE as a function of the compression rate r. Dots are obtained by training the decoder matrix $\boldsymbol{A}$ and the parameters $(\alpha_1, \alpha_2, \alpha_3)$ via SGD on masked ($p = 0.4$, green) or original ($p = 1$, blue) CIFAR-10 images. Continuous lines refer to the predictions of Proposition 6.5.1 for the optimal $f = f^*$ in (6.18), where x_1 has a Laplace distribution ($p = 1$, blue) or a sparse Laplace distribution ($p = 0.4$, orange). These curves match well the corresponding values obtained via SGD. Orange dots are obtained by training the matrices $\boldsymbol{A}, \boldsymbol{B}$ and the parameters $(\alpha_1, \alpha_2, \alpha_3)$ via SGD when $\boldsymbol{x}$ has i.i.d. sparse Laplace entries with $p = 0.4$.

a Haar matrix, and it is fixed; the decoder matrix $\boldsymbol{A}$ and the parameters $(\alpha_1, \alpha_2, \alpha_3)$ in the definition (6.15) of f are obtained via SGD training. Two observations are in order:

1. The autoencoder in (6.4) captures the sparsity: the MSE achieved on sparse data ($p = 0.4$, green dots) is lower than the MSE on non-sparse data ($p = 1$, blue dots).
2. For both values of p, the SGD performance matches the RHS of (6.16) (continuous lines) with $f = f^*$. As expected, this MSE is smaller than $1 - r \cdot (\mathbb{E}|x_1|)^2$, and it coincides with that obtained for compressing synthetic data with i.i.d. Laplace entries (orange dots).

6.5.2 Provable Benefit of Depth

We conclude by showing that the MSE can be further reduced by considering a multi-layer decoder. Our design of the decoding architecture is inspired by the RI-GAMP algorithm [VKM22], which iteratively estimates $\boldsymbol{x}$ from an observation of the form $\sigma(\boldsymbol{B}\boldsymbol{x})$ via

$$\begin{aligned}\boldsymbol{x}^t &= \boldsymbol{B}^\top \hat{\boldsymbol{z}}^t - \sum_{i=1}^{t-1} \beta_{t,i} \hat{\boldsymbol{x}}^i, \quad \hat{\boldsymbol{x}}^t = f_t(\boldsymbol{x}^1, \cdots, \boldsymbol{x}^t), \\ \boldsymbol{z}^t &= \boldsymbol{B}\hat{\boldsymbol{x}}^t - \sum_{i=1}^{t} \alpha_{t,i} \hat{\boldsymbol{z}}^i, \quad \hat{\boldsymbol{z}}^{t+1} = g_t(\boldsymbol{z}^1, \cdots, \boldsymbol{z}^t, \hat{\boldsymbol{z}}^1).\end{aligned} \tag{6.20}$$

Here, f_t, g_t are Lipschitz and applied component-wise, and the initialization is $\hat{\boldsymbol{z}}^1 = \mathrm{sign}(\boldsymbol{B}\boldsymbol{x})$. The coefficients $\{\beta_{t,i}\}$ and $\{\alpha_{t,i}\}$ are chosen so that, under suitable assumptions on $\boldsymbol{B}$,[3] the empirical distribution of the iterates is tracked via a low-dimensional recursion, known as *state evolution*. This in turn allows to evaluate the MSE $\lim_{d\to\infty} \frac{1}{d}\|\boldsymbol{x} - \hat{\boldsymbol{x}}^t\|_2^2$.

[3] $\boldsymbol{B}$ has to be bi-rotationally invariant in law, namely, the matrices appearing in its SVD are sampled from the Haar distribution.

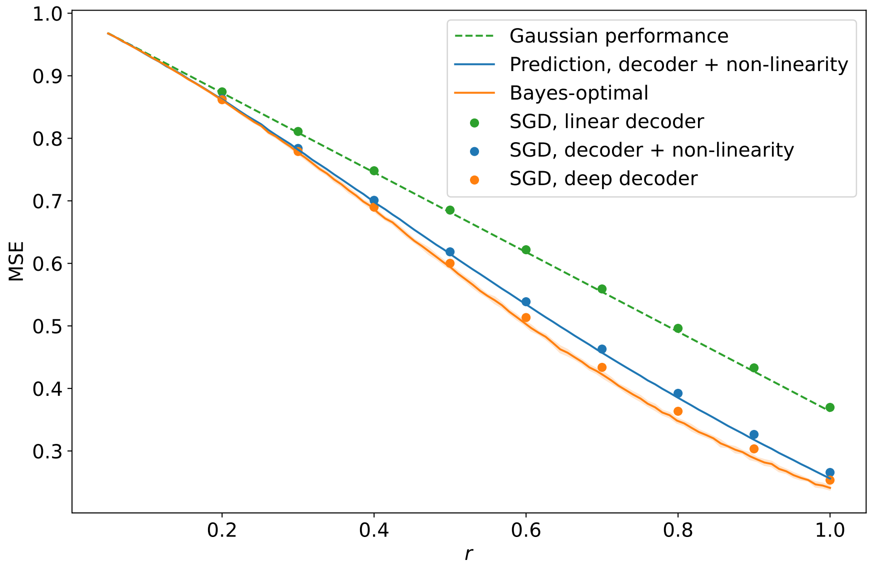


Figure 6.9: Compression of sparse Gaussian data $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ for $p = 0.3$ and $d = 500$. We plot the MSE as a function of the compression rate r for various autoencoder architectures. The architecture in (6.21) (orange dots) outperforms the autoencoders in (6.2) (green dots) and in (6.4) (blue dots), and it approaches the Bayes-optimal MSE (orange line).

The results of Proposition 6.4.1 and 6.5.1 follow from relating the autoencoders in (6.2)-(6.4) to RI-GAMP iterates in (6.20). More generally, $\hat{\boldsymbol{x}}^t$ is obtained by multiplications with $\boldsymbol{B}, \boldsymbol{B}^\top$, linear combinations of previous iterates, and component-wise applications of Lipschitz functions. As such, it can be expressed via a multi-layer decoder with residual connections. The numerical results in [VKM22] show that taking f_t, g_t as posterior means (as in (6.18)) leads to Bayes-optimal performance, having fixed the encoder matrix $\boldsymbol{B}$. Thus, this provides a proof-of-concept of the optimality of multi-layer decoders.

In fact, Figure 6.9 shows that an architecture with three decoding layers is already near-optimal when $\boldsymbol{x} \sim \mathrm{SG}_d(p)$. The decoder output is $\hat{\boldsymbol{x}}^2$ computed as (see also the block diagram in Figure 6.10)

$$\begin{aligned}
\hat{\boldsymbol{z}}^1 &= \mathrm{sign}(\boldsymbol{B}\boldsymbol{x}), \quad \boldsymbol{x}^1 = \boldsymbol{W}_1\hat{\boldsymbol{z}}^1, \quad \hat{\boldsymbol{x}}^1 = f_1(\boldsymbol{x}^1), \\
\hat{\boldsymbol{z}}^2 &= g_1(\boldsymbol{V}_1\hat{\boldsymbol{x}}^1 \oplus_1 \hat{\boldsymbol{z}}^1), \\
\boldsymbol{x}^2 &= \hat{\boldsymbol{x}}^1 \oplus_2 \boldsymbol{W}_2\hat{\boldsymbol{z}}^2, \quad \hat{\boldsymbol{x}}^2 = f_2(\boldsymbol{x}^1 \oplus_3 \boldsymbol{x}^2).
\end{aligned} \tag{6.21}$$

Here, $f_1(\cdot), f_2(\cdot), g_1(\cdot)$ are trainable parametric functions of the form in (6.15) and, for $i \in \{1, 2, 3\}$, $\boldsymbol{a} \oplus_i \boldsymbol{b} = \beta_i\boldsymbol{a} + \gamma_i\boldsymbol{b}$, where $\{\beta_i, \gamma_i\}$ are also trained. The plot demonstrates the benefit of employing more expressive decoders:

1. The green dots are obtained via SGD training of the autoencoder in (6.2) and, as proved in Theorem 9, they match the Gaussian performance $\mathcal{R}_{\mathrm{Gauss}}$.
2. The blue dots are obtained via SGD training of the autoencoder in (6.4) and they match the prediction of Proposition 6.5.1 with $f = f^*$ in (6.18).
3. The orange dots are obtained by using the decoder in (6.21) where $\boldsymbol{W}_1 = \boldsymbol{W}_2 = \boldsymbol{B}^\top$, $\boldsymbol{V}_1 = \boldsymbol{B}$ are subsampled Haar matrices and the parameters in the functions $f_1, f_2, g_1, \{\oplus_i\}_{i=1}^3$ are trained via SGD. Similar results are obtained by training also $\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{V}_1$, although at the cost of a slower convergence.

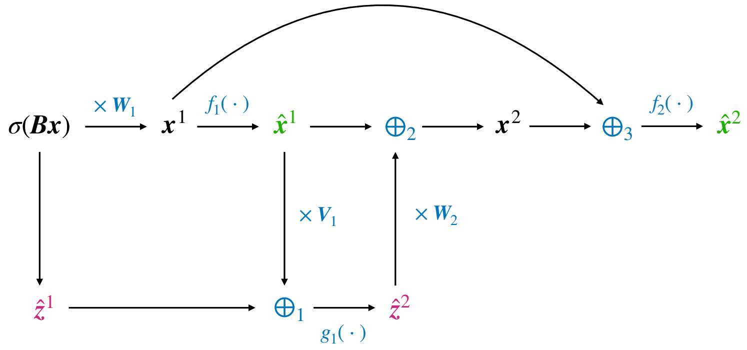


Figure 6.10: Block diagram of the decoder in (6.21).

In summary, the architecture in (6.21) improves upon those in (6.2)-(6.4), and it approaches the orange curve which gives the Bayes-optimal MSE achievable by fixing a rotationally invariant encoder matrix $\boldsymbol{B}$ [MXM21]. Additional details are deferred to Appendix D.2.7.

We also note that considering a deep fully-connected decoder in place of the architecture in (6.21) does not improve upon the autoencoder in (6.4). In fact, while sufficiently wide and deep models have high expressivity, their SGD training is notoriously difficult, due to e.g. vanishing/exploding gradients [GB10, HZRS16]. In addition to that, we would like to point out that training with quantized activations (such as sign) which utilize variants of the straight-through estimator introduces challenges for the optimization. This has led to the extensive usage of heuristics to make training more stable, such as "clipping" (see, e.g., [HCS$^+$18]).

6.6 Conclusions and Future Directions

Let us summarize the key points of our analysis presented in this chapter. Motivated by the Gaussian universality of the (shallow) two-layer model (6.2) demonstrated on MNIST and CIFAR-10 data in Chapter 5, we studied the behaviour of the shallow model on a more structured sparse Gaussian source. Our analysis (Theorem 9) of the gradient-based optimization scheme unveils that the shallow autoencoder model is incapable of capturing the structure of the data (sparsity) and recovers the Gaussian performance (6.5). Going beyond sparse Gaussian data, we derive a conjecture for a general i.i.d. input distribution. The conjecture states that the shallow autoencoder differentiates between two parameter configurations (6.10): the sparse deterministic permutation of identity and the random Haar design. The condition which guides this choice corresponds to the comparison between the first absolute moment of the input distribution and the analogous quantity evaluated for Gaussian signal (Proposition 6.4.1). We empirically validate that the aforementioned holds for more general data: MNIST and CIFAR-10 natural images, and particle physics data [YM21b]. Moreover, when the transition from random to deterministic design takes place, the SGD training dynamics (see, for instance, Figure 6.3) shows a "staircase" behaviour.

In Section 6.5.1, we indicate that a similar behaviour is observed for a more expressive decoding model (6.4) that utilizes an extra non-linearity for general i.i.d. priors. In Section 6.5.2, we conclude with an AMP-based neural decoder architecture that is able to achieve performance close to the Bayes optimal one (given the rotationally invariant Haar design) at the cost of only three extra layers and parametric non-linearities. We also indicate that the "intelligent" design of the decoder is essential as empirically a "straight-forward" MLP decoder does not improve upon a single non-linearity (6.4).

The above summary quite evidently suggests a number of exciting directions for future research. First, proving formally the observed "staircase" phenomenon would be an interesting and challenging result by itself, since the technical machinery, developed for the analysis in Theorems 9 and 7, is no longer applicable to the second "sparse" phase, due to a lack of the concentration provided by the Haar design. Thus, tackling this problem will require the development of new technical machinery, which will be of separate interest.

Another interesting direction corresponds to analysing more rich encoding schemes, since the GLM design of the encoder is clearly sub-optimal even in the presence of the best possible decoder (compare Figures 6.9 and 6.1 for the sparse Gaussian case, and see Figure 5.3 for Gaussian data). Furthermore, for the theoretical analysis we consider data with $\mathrm{i.i.d.}$ components. Consequently, it would be interesting to obtain results similar to the non-isotropic Gaussian case presented in Chapter 5 for a more general class of inputs with correlated components. For instance, we expect similar in flavour results might be obtained (with enough effort) for the following class of inputs

$$\boldsymbol{x} = \boldsymbol{P} \cdot \widetilde{\boldsymbol{x}}, \quad \boldsymbol{P} \in \mathbb{R}^{d \times d}, \quad \widetilde{\boldsymbol{x}} \in \mathbb{R}^{d},$$

for some PSD matrix $\boldsymbol{P}$ and where the $\mathrm{i.i.d.}$ "latent" $\widetilde{\boldsymbol{x}}$ has some structure, i.e., sparsity. A more exciting (albeit more challenging) direction would be to consider data with "effective" dimension less than d. To be more concrete, one may consider the "hidden manifold" data model (see, for example, [GMKZ20]) which could be formalized as the following general relation:

$$\boldsymbol{x} = \psi(\widetilde{\boldsymbol{x}}), \quad \widetilde{\boldsymbol{x}} \in \mathbb{R}^{d'}, \quad d' < d,$$

where $\psi : \mathbb{R}^{d'} \to \mathbb{R}^{d}$ is some (possibly non-linear) mapping from the lower-dimensional latent representation of $\boldsymbol{x}$ to the space of inputs. In particular, it would be interesting to derive any quantitative result which suggests that a suitable autoencoder model is able to, vaguely speaking, capture the lower-dimensional structure of the signal. For instance, a satisfactory analysis would imply that the population risk bounds akin to the ones described in Theorem 5 and Proposition 5.4.2 would implicitly depend not on the ambient dimension d but on the dimension of the "latent" d' (and most likely on some complexity measure of the transformation ψ itself).

CHAPTER 7

Discussion and Concluding Remarks

In this thesis, we study the phenomenology that emerges in artificial neural networks under various asymptotic regimes. Utilizing "high-dimensions", we provably show that there are a few properties specific to over-parameterized models and, armed with the asymptotic analysis, provide a sharp characterization of the neural network behaviour.

Thesis Summary and Contributions

In Chapter 3 and 4, we take the mean-field view [MMN18, AOY19] to study over-parameterized neural networks in the supervised learning (regression and classification) context. Inspired by the empirically observed phenomenon of mode connectivity [GIP+18, DVSH18] and the related theoretical property of dropout stability [KWL+19], in Chapter 3 we provide a rigorous proof along with quantitative bounds for this observation. Crucially, the analysis heavily relies on the over-parameterization of the neural architecture at hand. In this view, our results also provide a theoretical basis for the practical success of the family of "local search" algorithms, such as stochastic gradient descent (SGD), as in this case the model's optimization landscape is more well-behaved due to mode connectivity, contrary to what the worst case scenario might suggest.

The practical success of gradient-based methods is sometimes attributed to the implicit bias of the optimization procedure itself. Namely, it is conjectured and proven in certain cases (e.g., [WTS+19, BGVV20]) that the learning algorithm is implicitly selective, i.e., it finds functionally simple solutions that exhibit superior generalization ability in comparison to other candidates with roughly the same value of the empirical risk. In Chapter 4, we capitalize on the mean-field characterization of the training dynamics once more to show that SGD on an over-parameterized ReLU neural network is attracted to a relatively simple solution. In particular, the network implements a piecewise linear function: the number of tangent changes is independent of the network size and scales linearly with the number of training samples. Remarkably, the described behaviour is significantly different from related works [WTS+19, BGVV20]. In addition to that our proof technique directly utilizes the functional form of the Gibbs minimizer. In this view, the fact that it allows for such a tight analysis is quite surprising and might be of separate interest to the community.

In Chapter 5 and 6, we divert from the supervised setting and analyse the autoencoding paradigm that showed remarkable success on various unsupervised tasks such as representation learning [TBL18] and generative modeling [KW13]. Despite their wide practical spread, the theoretical understanding of such models is quite limited even in the simplistic shallow case. Precisely, the existing analysis is either restricted to linear autoencoders which yield PCA-like behaviour [KBGS19, OSWS20], or extreme compression rate regimes [Ngu21, RG22]. In Chapter 5, we aim to bridge this gap and consider a shallow autoencoder (AE) for 1-bit compression of Gaussian inputs in the challenging proportional regime (the compression rate is fixed to a constant, i.e., it is neither vanishing nor diverging in contrast to the results described in [Ngu21, RG22]). We provide lower bounds on the reconstruction error of the AE under such setting. Importantly, the tightness of the bounds is once more ensured by the "blessing" of high-dimensions. It also allows to provide rigorous guarantees for the convergence of a certain gradient-based scheme to the global minimizer. Surprisingly, we discover that the predictions made under the Gaussian assumption translate extremely well to the case of natural data, such as MNIST and CIFAR-10 images.

In Chapter 6, we investigate this "universality" property further. In particular, we ask the question whether the data is indeed more Gaussian than it appears or it is the specific flaw of the shallow design which disables the model from seeing beyond such crude approximation of the input signal. To start with, we consider a prototypical example of a more "structured" signal - sparse Gaussian data - which at least information-theoretically is more amenable to compression. However, we prove that the gradient-based scheme fails to recover the sparsity of the inputs, and that at convergence a shallow autoencoder model is incapable to capture the structure, which is explained by the resulting Gaussian performance. Digging into the issue deeper, we focus on the case of general $\mathrm{i.i.d.}$ source. We conjecture that depending on a certain data statistic the model chooses between two candidates of different but equally simplistic nature. Namely, the optimal encoder design is either rotationally invariant or deterministic and sparse (permutation of identity). Notably, when the rotationally invariant encoding is not the optimal choice, the training dynamics of the autoencoder exhibits a "staircase" behaviour [ABAB$^+$21, AAM22, AAM23, PF23]. We also provide both empirical and theoretical evidence that the "curse" of Gaussian performance can be alleviated by enriching the decoding architecture.

Future Directions

In this thesis we extensively demonstrated that taking a high-dimensional view is beneficial from both analytical, i.e., making the problem at hand tractable, and phenomenological perspectives, i.e., that certain properties (e.g., dropout stability) occur naturally under sufficient over-parameterization. However, there are a few fundamental questions related to the results presented in the current thesis which remain unanswered, especially concerning the second part of the thesis focused on the autoencoding paradigm. In the following paragraphs, we summarize the future directions which were previously discussed in the corresponding sections of the current thesis and also mention a few (mostly on the technical side) that were omitted.

Loss landscape and implicit bias. We start by highlighting a few directions in which the results presented in Chapter 3 can be extended. In particular, we focus on the remarks concerning the multi-layer case presented in Section 3.4. First, we would like to note that the restriction on the first and last layer parameters to stay fixed during the SGD training for the theoretical analysis could be alleviated by considering a more recent and refined version of the corresponding mean-field coupling [NP23]. However, let us point out that incorporating bias terms in the intermediate layers might still prove to be a challenge. This stems from the fact that in the absence of bias terms the corresponding limit admits significant simplifications (see the more detailed discussion in Section 5 of [NP23]). Let us also briefly mention that the exponential in time coupling in the main theorems of Chapter 3 could potentially be fixed via novel uniform in time propagation of chaos results, see, for example, [SNW23].

With regards to the implicit bias results presented in Chapter 4, the natural direction of the extension corresponds to translating the current analysis to the case of multidimensional regression. However, the straight-forward way would imply a number of "knots" which is exponential in the input dimension. In this case, "knots" are multidimensional and correspond to the intersection of the hyperplanes which define the activation region of each ReLU neuron. Thus, we do not expect to see a relatively trivial extension without the need to change the nature of the argument itself or adding extra assumptions on the input data. On more technical side, let us remark that the uniqueness of the limit on the approximating sequence (i.e., that the piecewise linear solution structure does not change by selecting a different subsequence) is left without proof. However, we do present sufficient empirical evidence that the uniqueness in fact holds. Establishing the formal proof might require considerable effort since morally it would correspond to providing an extra "continuity" argument (e.g., uniform integrability on the sequence).

Autoencoders and feature learning. Let us now proceed with highlighting possible future directions for the analysis of autoencoders in Chapters 5 and 6. To start, given the direct comparison with the shallow model (see, for instance, Figure 5.3), it is quite evident that the shallow architecture is quite far off from the optimal encoding-decoding scheme. In this view, it would be exciting to find a suitable deeper neuronal architecture which could get closer or, best case, saturate the rate-distortion prediction.

It would be also interesting to obtain quantitative results akin to the lower bounds discussed in Chapter 5 for the case of more structured input data. Namely, assuming that the source signal is supported on a certain lower-dimensional space, is it possible to derive the population risk bound which would explicitly depend on this effective dimension in a non-trivial way? Along the same lines but more towards explicitly quantifying the "feature learning" aspect of the autoencoders, given a higher order structural correlation in the inputs (e.g., triplets of input coordinates have a strong dependency) which features does the encoding extract from the data? In the case of signals considered in Chapters 5 and 6 (lower order) the features themselves are not very informative (either Haar matrix or no features at all, which corresponds to the permutation of identity), except the case with covariance, discussed in Section 5.5, when the encoding learns the correct eigenspaces of the data. For a more technical discussion, we also refer the reader to Section 6.6 of the current thesis.

In conclusion, in this thesis we explored how high-dimensional regimes both naturally explain the emergent properties in modern machine learning systems while simultaneously paving the way for the related theoretical analysis. We hope that the results presented in the current thesis together with the developed methodology would give a new perspective on understanding practical high-dimensional regimes for large scale artificial neural networks, and provide novel theoretical machinery for analysing many particle systems in general.

Bibliography

[AAM22] Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for SGD learning of sparse functions on two-layer neural networks. *Conference on Learning Theory*, 2022.

[AAM23] Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. SGD learning on neural networks: leap complexity and saddle-to-saddle dynamics. *Conference on Learning Theory*, 2023.

[ABAB$^+$21] Emmanuel Abbe, Enric Boix-Adsera, Matthew S Brennan, Guy Bresler, and Dheeraj Nagaraj. The staircase property: How hierarchical structure can guide deep learning. *Advances in Neural Information Processing Systems*, 2021.

[ABBA$^+$23] Emmanuel Abbe, Samy Bengio, Enric Boix-Adserà, Etai Littwin, and Joshua M Susskind. Transformers learn through gradual rank increase. *Advances in Neural Information Processing Systems*, 2023.

[AHW96] Peter Auer, Mark Herbster, and Manfred KK Warmuth. Exponentially many local minima for single neurons. *Advances in Neural Information Processing Systems*, 1996.

[AMS09] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[AMT$^+$17] Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc V Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. *Advances in Neural Information Processing Systems*, 2017.

[AOY19] Dyego Araújo, Roberto I Oliveira, and Daniel Yukimura. A mean-field limit for certain deep neural networks. *arXiv preprint*, 2019.

[Ari72] S. Arimoto. An algorithm for computing the capacity of arbitrary discrete memoryless channels. *IEEE Transactions on Information Theory*, 1972.

[AZLL19] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *Advances in Neural Information Processing Systems*, 2019.

[AZLS19] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. *International Conference on Machine Learning*, 2019.

[Bar93] Andrew R Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information theory*, 1993.

[Bar98] Peter Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, 1998.

[BB08] Petros T Boufounos and Richard G Baraniuk. 1-bit compressive sensing. *Conference on Information Sciences and Systems*, 2008.

[BB18] Randall Balestriero and Richard Baraniuk. A spline theory of deep learning. *International Conference on Machine Learning*, 2018.

[BBV04] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[BCB14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*, 2014.

[BCM$^+$21] Johannes Ballé, Philip A. Chou, David Minnen, Saurabh Singh, Nick Johnston, Eirikur Agustsson, Sung Jin Hwang, and George Toderici. Nonlinear transform coding. *IEEE Transactions on Special Topics in Signal Processing*, 2021.

[BDS19] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *International Conference on Learning Representations*, 2019.

[Ber23] Raphaël Berthier. Incremental learning in diagonal linear networks. *Journal of Machine Learning Research*, 2023.

[BGVV20] Guy Blanc, Neha Gupta, Gregory Valiant, and Paul Valiant. Implicit regularization for deep neural networks driven by an Ornstein-Uhlenbeck like process. *Conference on Learning Theory*, 2020.

[BH89] Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 1989.

[BHMM19] 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*, 2019.

[BKM$^+$19] Jean Barbier, Florent Krzakala, Nicolas Macris, Léo Miolane, and Lenka Zdeborová. Optimal errors and phase transitions in high-dimensional generalized linear models. *Proceedings of the National Academy of Sciences*, 2019.

[Bla72] R. Blahut. Computation of channel capacity and rate-distortion functions. *IEEE Transactions on Information Theory*, 1972.

[BLLT20] Peter Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 2020.

[BLS17] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli. End-to-end optimized image compression. *International Conference on Learning Representations*, 2017.

[BLSG20] Xuchan Bao, James Lucas, Sushant Sachdeva, and Roger B Grosse. Regularized linear autoencoders recover the principal components, eventually. *Advances in Neural Information Processing Systems*, 2020.

[BM11] Mohsen Bayati and Andrea Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 2011.

[BMR21] Peter Bartlett, Andrea Montanari, and Alexander Rakhlin. Deep learning: A statistical viewpoint. *Acta Numerica*, 2021.

[BMZ23] Raphaël Berthier, Andrea Montanari, and Kangjie Zhou. Learning time-scales in two-layers neural networks. *arXiv preprint*, 2023.

[BR89] Avrim Blum and Ronald L Rivest. Training a 3-node neural network is NP-complete. *Advances in Neural Information Processing Systems*, 1989.

[BVB16] Nicolas Boumal, Vlad Voroninski, and Afonso Bandeira. The non-convex burer-monteiro approach works on smooth semidefinite programs. *Advances in Neural Information Processing Systems*, 2016.

[BYAV13] Yoshua Bengio, Li Yao, Guillaume Alain, and Pascal Vincent. Generalized denoising auto-encoders as generative models. *Advances in Neural Information Processing Systems*, 2013.

[CB18] Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. *Advances in Neural Information Processing Systems*, 2018.

[CB20] Lenaic Chizat and Francis Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. *Conference on Learning Theory*, 2020.

[CCGZ20] Zixiang Chen, Yuan Cao, Quanquan Gu, and Tong Zhang. Mean-field analysis of two-layer neural networks: Non-asymptotic rates and generalization bounds. *arXiv preprint*, 2020.

[CFW$^+$21] Yuan Cao, Zhiying Fang, Yue Wu, Ding-Xuan Zhou, and Quanquan Gu. Towards understanding the spectral bias of deep learning. *International Joint Conference on Artificial Intelligence*, 2021.

[Chi22] Lénaïc Chizat. Mean-field langevin dynamics: Exponential convergence and annealing. *Transactions on Machine Learning Research*, 2022.

[CHM$^+$15] Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. *International Conference on Artificial Intelligence and Statistics*, 2015.

[CLS15] Emmanuel J Candes, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 2015.

[CMZ06] Stefano Ciliberti, Marc Mézard, and Riccardo Zecchina. Message-passing algorithms for non-linear nodes and data compression. *ComPlexUs*, 2006.

[COB19] L. Chizat, E. Oyallon, and F. Bach. On lazy training in differentiable programming. *Advances in Neural Information Processing Systems*, 2019.

[CSTK20] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. *Conference on Computer Vision and Pattern Recognition*, 2020.

[CSV13] Emmanuel J Candes, Thomas Strohmer, and Vladislav Voroninski. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Communications on Pure and Applied Mathematics*, 2013.

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 1995.

[Cyb89] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 1989.

[CZ23] Hugo Cui and Lenka Zdeborová. High-dimensional asymptotics of denoising autoencoders. *Advances in Neural Information Processing Systems*, 2023.

[DBDFS20] Valentin De Bortoli, Alain Durmus, Xavier Fontaine, and Umut Simsekli. Quantitative propagation of chaos for SGD in wide neural networks. *Advances in Neural Information Processing Systems*, 2020.

[DBK+21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations*, 2021.

[DDS+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *Conference on Computer Vision and Pattern Recognition*, 2009.

[DLL+18] Simon S Du, Jason D Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. *International Conference on Machine Learning*, 2018.

[DMM09] David L Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 2009.

[dPG95] Guido E del Pino and Hector Galaz. Statistical applications of the inverse gram matrix: A revisitation. *Brazilian Journal of Probability and Statistics*, 1995.

[DPG+14] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *Advances in Neural Information Processing Systems*, 2014.

[DPS20] Alex Dytso, H Vincent Poor, and Shlomo Shamai Shitz. A general derivative identity for the conditional mean estimator in gaussian noise and some applications. *IEEE International Symposium on Information Theory*, 2020.

[DSL22] Rishabh Dudeja, Subhabrata Sen, and Yue M Lu. Spectral universality of regularized linear regression with nearly deterministic sensing matrices. *arXiv preprint*, 2022.

[DVSH18] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. *International Conference on Machine Learning*, 2018.

[DZPS19] Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *International Conference on Learning Representations*, 2019.

[EP21] Tolga Ergen and Mert Pilanci. Convex geometry and duality of over-parameterized neural networks. *Journal of Machine Learning Research*, 2021.

[EP23] Tolga Ergen and Mert Pilanci. The convex landscape of neural networks: Characterizing global optima and stationary points via lasso models. *arXiv preprint*, 2023.

[FB17] C Daniel Freeman and Joan Bruna. Topology and geometry of half-rectified network optimization. *International Conference on Learning Representations*, 2017.

[FC19] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *International Conference on Learning Representations*, 2019.

[FLYZ21] Cong Fang, Jason Lee, Pengkun Yang, and Tong Zhang. Modeling from features: a mean-field framework for over-parameterized deep neural networks. *Conference on Learning Theory*, 2021.

[FR18] Alyson K Fletcher and Sundeep Rangan. Iterative reconstruction of rank-one matrices in noise. *Information and Inference: A Journal of the IMA*, 2018.

[FRS18] Alyson K Fletcher, Sundeep Rangan, and Philip Schniter. Inference in deep networks in high dimensions. *IEEE International Symposium on Information Theory*, 2018.

[FSARS16] Alyson Fletcher, Mojtaba Sahraee-Ardakan, Sundeep Rangan, and Philip Schniter. Expectation consistent approximate inference: Generalizations and convergence. In *IEEE International Symposium on Information Theory*, 2016.

[Fun89] Ken-Ichi Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 1989.

[FVR$^+$22] Oliver Y Feng, Ramji Venkataramanan, Cynthia Rush, Richard J Samworth, et al. A unifying tutorial on approximate message passing. *Foundations and Trends® in Machine Learning*, 2022.

[GB10] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. *International Conference on Artificial Intelligence and Statistics*, 2010.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[GBEK04] Véronique Gayrard, Anton Bovier, Michael Eckhoff, and Markus Klein. Metastability in reversible diffusion processes I: Sharp asymptotics for capacities and exit times. *Journal of the European Mathematical Society*, 2004.

[GBLJ19] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. *Advances in Neural Information Processing Systems*, 2019.

[GEH19] Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks. *arXiv preprint*, 2019.

[GIP$^+$18] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in Neural Information Processing Systems*, 2018.

[GJZ17] Rong Ge, Chi Jin, and Yi Zheng. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. *International Conference on Machine Learning*, 2017.

[GLM16] Rong Ge, Jason Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. *Advances in Neural Information Processing Systems*, 2016.

[GLQ$^+$19] Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General analysis and improved rates. *International Conference on Machine Learning*, 2019.

[GLR$^+$22] Sebastian Goldt, Bruno Loureiro, Galen Reeves, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. The gaussian equivalence of generative models for learning with shallow neural networks. *Mathematical and Scientific Machine Learning*, 2022.

[GMKZ20] Sebastian Goldt, Marc Mézard, Florent Krzakala, and Lenka Zdeborová. Modeling the influence of data structure on learning in neural networks: The hidden manifold model. *Physical Review X*, 2020.

[GMZ09] Bogdan Grechuk, Anton Molyboha, and Michael Zabarankin. Maximum entropy principle with general deviation measures. *Mathematics of Operations Research*, 2009.

[GPAM$^+$14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Advances in Neural Information Processing Systems*, 2014.

[Gra84] Robert Gray. Vector quantization. *IEEE ASSP Magazine*, 1984.

[GVS15] Ian Goodfellow, Oriol Vinyals, and Andrew Saxe. Qualitatively characterizing neural network optimization problems. *International Conference on Learning Representations*, 2015.

[GYC16] Yiwen Guo, Anbang Yao, and Yurong Chen. Dynamic network surgery for efficient DNNs. *Advances In Neural Information Processing Systems*, 2016.

[HCS$^+$18] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *Journal of Machine Learning Research*, 2018.

[HJA20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020.

[HL22] Hong Hu and Yue M Lu. Universality laws for high-dimensional learning with random features. *IEEE Transactions on Information Theory*, 2022.

[HMRT22] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *Annals of Statistics*, 2022.

[HR11] Tien-Chung Hu and Andrew Rosalsky. A note on the de La Vallée Poussin criterion for uniform integrability. *Statistics & Probability Letters*, 2011.

[HRŠS21] Kaitong Hu, Zhenjie Ren, David Šiška, and Łukasz Szpruch. Mean-field langevin dynamics and energy landscape of neural networks. *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, 2021.

[HS06] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.

[HWJ17] Hengtao He, Chao-Kai Wen, and Shi Jin. Generalized expectation consistent signal recovery for nonlinear measurements. *IEEE International Symposium on Information Theory*, 2017.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Conference on Computer Vision and Pattern Recognition*, 2016.

[JEP$^+$21] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 2021.

[JGH18] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in Neural Information Processing Systems*, 2018.

[JGȘ$^+$21] Arthur Jacot, François Ged, Berfin Șimșek, Clément Hongler, and Franck Gabriel. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity. *arXiv preprint*, 2021.

[JKO98] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the Fokker–Planck equation. *SIAM Journal on Mathematical Analysis*, 1998.

[JM23] Hui Jin and Guido Montúfar. Implicit bias of gradient descent for mean squared error regression with wide neural networks. *Journal of Machine Learning Research*, 2023.

[JMM20] Adel Javanmard, Marco Mondelli, and Andrea Montanari. Analysis of a two-layer neural network via displacement convexity. *Annals of Statistics*, 2020.

[JRU21] Saachi Jain, Adityanarayanan Radhakrishnan, and Caroline Uhler. A mechanism for producing aligned latent spaces with autoencoders. *arXiv preprint*, 2021.

[Kaw16] Kenji Kawaguchi. Deep learning without poor local minima. *Advances in Neural Information Processing Systems*, 2016.

[KBGS19] Daniel Kunin, Jonathan Bloom, Aleksandrina Goeva, and Cotton Seed. Loss landscapes of regularized linear autoencoders. *International Conference on Machine Learning*, 2019.

[Kel03] Carl T Kelley. *Solving nonlinear equations with Newton's method*. SIAM, 2003.

[Kha21] Apoorva Khare. Sharp nonzero lower bounds for the schur product theorem. *Proceedings of the American Mathematical Society*, 2021.

[KMN$^+$17] 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. *International Conference on Learning Representations*, 2017.

[KSH17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 2017.

[KSHM24] Kevin Kögler, Alexander Shevchenko, Hamed Hassani, and Marco Mondelli. Compression of structured data with autoencoders: provable benefit of nonlinearities and depth. *International Conference on Machine Learning*, 2024.

[KU10] Satish Babu Korada and Rüdiger L Urbanke. Polar codes are optimal for lossy source coding. *IEEE Transactions on Information Theory*, 2010.

[KW13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint*, 2013.

[KW14] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014.

[KWL$^+$19] Rohith Kuditipudi, Xiang Wang, Holden Lee, Yi Zhang, Zhiyuan Li, Wei Hu, Sanjeev Arora, and Rong Ge. Explaining landscape connectivity of low-cost solutions for multilayer nets. *Advances in Neural Information Processing Systems*, 2019.

[Lau15] Philippe Laurençot. Weak compactness techniques and coagulation equations. *Evolutionary Equations with Applications in Natural Sciences*, 2015.

[LBBH98] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

[LCG$^+$19] Haojie Liu, Tong Chen, Peiyao Guo, Qiu Shen, Xun Cao, Yao Wang, and Zhan Ma. Non-local attention optimized deep image compression. *arXiv preprint*, 2019.

[LFSW23] Yufan Li, Zhou Fan, Subhabrata Sen, and Yihong Wu. Random linear estimation with rotationally-invariant designs: Asymptotics at high temperature. *IEEE Transactions on Information Theory*, 2023.

[LGC$^+$21] Bruno Loureiro, Cedric Gerbelot, Hugo Cui, Sebastian Goldt, Florent Krzakala, Marc Mezard, and Lenka Zdeborová. Learning curves of generic features maps for realistic datasets with a teacher-student model. *Advances in Neural Information Processing Systems*, 2021.

[LHB22] Eric Lei, Hamed Hassani, and Shirin Saeedi Bidokhti. Neural estimation of the rate-distortion function for massive datasets. *IEEE International Symposium on Information Theory*, 2022.

[LL18] Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *Advances in Neural Information Processing Systems*, 2018.

[LL20] Yulong Lu and Jianfeng Lu. A universal approximation theorem of deep neural networks for expressing probability distributions. *Advances in Neural Information Processing Systems*, 2020.

[LSLS18a] Shiyu Liang, Ruoyu Sun, Jason D Lee, and Rayadurgam Srikant. Adding one neuron can eliminate all bad local minima. *Advances in Neural Information Processing Systems*, 2018.

[LSLS18b] Shiyu Liang, Ruoyu Sun, Yixuan Li, and Rayadurgam Srikant. Understanding the loss surface of neural networks for binary classification. *International Conference on Machine Learning*, 2018.

[LSSS14] Roi Livni, Shai Shalev-Shwartz, and Ohad Shamir. On the computational efficiency of training neural networks. *Advances in Neural Information Processing Systems*, 2014.

[LSZ$^+$19] Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. *International Conference on Learning Representations*, 2019.

[LZG21] Zhu Li, Zhi-Hua Zhou, and Arthur Gretton. Towards an understanding of benign overfitting in neural networks. *arXiv preprint*, 2021.

[MAV17] Dmitry Molchanov, Arsenii Ashukha, and Dmitry Vetrov. Variational dropout sparsifies deep neural networks. *International Conference on Machine Learning*, 2017.

[MBG18] Hartmut Maennel, Olivier Bousquet, and Sylvain Gelly. Gradient descent quantizes ReLU network features. *arXiv preprint*, 2018.

[MBM18] Song Mei, Yu Bai, and Andrea Montanari. The landscape of empirical risk for non-convex losses. *Annals of Statistics*, 2018.

[Mec19] Elizabeth S Meckes. *The random matrix theory of the classical compact groups*, volume 218. Cambridge University Press, 2019.

[Min01] Thomas P Minka. Expectation propagation for approximate bayesian inference. *Conference on Uncertainty in Artificial Intelligence*, 2001.

[MKAA21] Paolo Milanesi, Hachem Kadri, Stéphane Ayache, and Thierry Artières. Implicit regularization in deep tensor factorization. *IEEE International Joint Conference on Neural Networks*, 2021.

[MM22] Namiko Matsumoto and Arya Mazumdar. Binary iterative hard thresholding converges with optimal number of measurements for 1-bit compressed sensing. *IEEE Annual Symposium on Foundations of Computer Science*, 2022.

[MMM19] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. *Conference on Learning Theory*, 2019.

[MMN18] 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*, 2018.

[MN89] P. McCullagh and J. A. Nelder. Generalized linear models. *Monographs on Statistics and Applied Probability*, 1989.

[MP43] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 1943.

[MP17] Junjie Ma and Li Ping. Orthogonal amp. *IEEE Access*, 2017.

[MS22] Andrea Montanari and Basil N Saeed. Universality of empirical risk minimization. *Conference on Learning Theory*, 2022.

[MT13] Ryosuke Matsushita and Toshiyuki Tanaka. Low-rank matrix reconstruction and clustering via approximate message passing. *Advances in Neural Information Processing Systems*, 2013.

[Mur22] Kevin P Murphy. *Probabilistic machine learning: an introduction*. MIT press, 2022.

[MV21] Andrea Montanari and Ramji Venkataramanan. Estimation of low-rank matrices via approximate message passing. *Annals of Statistics*, 2021.

[MV22] Marco Mondelli and Ramji Venkataramanan. Approximate message passing with spectral initialization for generalized linear models. *Journal of Statistical Mechanics: Theory and Experiment*, 2022.

[MXM21] Junjie Ma, Ji Xu, and Arian Maleki. Analysis of sensing spectral for signal recovery under a generalized linear model. *Advances in Neural Information Processing Systems*, 2021.

[Ngu19a] Phan-Minh Nguyen. Mean field limit of the learning dynamics of multilayer neural networks. *arXiv preprint*, 2019.

[Ngu19b] Quynh Nguyen. On connected sublevel sets in deep learning. *International Conference on Machine Learning*, 2019.

[Ngu21] Phan-Minh Nguyen. Analysis of feature learning in weight-tied autoencoders via the mean field lens. *arXiv preprint*, 2021.

[NH17] Quynh Nguyen and Matthias Hein. The loss surface of deep and wide neural networks. *International Conference on Machine Learning*, 2017.

[NH18] Quynh Nguyen and Matthias Hein. Optimization landscape and expressivity of deep CNNs. *International Conference on Machine Learning*, 2018.

[NKB$^+$20] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *International Conference on Learning Representations*, 2020.

[NLB$^+$19] Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. Towards understanding the role of over-parametrization in generalization of neural networks. *International Conference on Learning Representations*, 2019.

[NMH19] Quynh Nguyen, Mahesh Chandra Mukkamala, and Matthias Hein. On the loss landscape of a class of deep neural networks with no bad local valleys. *International Conference on Learning Representations*, 2019.

[NP21] Phan-Minh Nguyen and Huy Tuan Pham. Global convergence of three-layer neural networks in the mean field regime. *International Conference on Learning Representations*, 2021.

[NP23] Phan-Minh Nguyen and Huy Tuan Pham. A rigorous framework for the mean field limit of multilayer neural networks. *Mathematical Statistics and Learning*, 2023.

[NTS15] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *Workshop Contribution at International Conference on Learning Representations*, 2015.

[NWH19] Thanh V Nguyen, Raymond KW Wong, and Chinmay Hegde. On the dynamics of gradient descent for autoencoders. *International Conference on Artificial Intelligence and Statistics*, 2019.

[NWH21] Thanh V Nguyen, Raymond KW Wong, and Chinmay Hegde. Benefits of jointly training autoencoders: An improved neural tangent kernel analysis. *IEEE Transactions on Information Theory*, 2021.

[NWS22] Atsushi Nitanda, Denny Wu, and Taiji Suzuki. Convex analysis of the mean field langevin dynamics. *International Conference on Artificial Intelligence and Statistics*, 2022.

[O'D14] Ryan O'Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.

[OSWS20] Reza Oftadeh, Jiayi Shen, Zhangyang Wang, and Dylan Shell. Eliminating the invariance on the loss landscape of linear autoencoders. *International Conference on Machine Learning*, 2020.

[OWJ05] Manfred Opper, Ole Winther, and Michael J Jordan. Expectation consistent approximate inference. *Journal of Machine Learning Research*, 2005.

[OWSS20] Greg Ongie, Rebecca Willett, Daniel Soudry, and Nathan Srebro. A function space view of bounded norm infinite width ReLU nets: The multivariate case. *International Conference on Learning Representations*, 2020.

[PB17] Jeffrey Pennington and Yasaman Bahri. Geometry of neural network loss surfaces via random matrix theory. *International Conference on Machine Learning*, 2017.

[PF23] Scott Pesme and Nicolas Flammarion. Saddle-to-saddle dynamics in diagonal linear networks. *Advances in Neural Information Processing Systems*, 2023.

[PIVA21] Alexandra Peste, Eugenia Iofinova, Adrian Vladu, and Dan Alistarh. Ac/dc: Alternating compressed/decompressed training of deep neural networks. *Advances in Neural Information Processing Systems*, 2021.

[PN20a] Rahul Parhi and Robert Nowak. Banach space representer theorems for neural networks and ridge splines. *Journal of Machine Learning Research*, 2020.

[PN20b] Rahul Parhi and Robert D Nowak. The role of neural network activation functions. *IEEE Signal Processing Letters*, 2020.

[Rad08] Vicentiu Radulescu. Rodrigues-type formulae for hermite and laguerre polynomials. *Analele Stiintifice ale Universitatii Ovidius Constanta*, 2008.

[Ran11] Sundeep Rangan. Generalized approximate message passing for estimation with random linear mixing. *IEEE International Symposium on Information Theory*, 2011.

[RBU20] Adityanarayanan Radhakrishnan, Mikhail Belkin, and Caroline Uhler. Overparameterized neural networks implement associative memory. *Proceedings of the National Academy of Sciences*, 2020.

[RDS$^+$15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

[RFC20] Alex Renda, Jonathan Frankle, and Michael Carbin. Comparing rewinding and fine-tuning in neural network pruning. *International Conference on Learning Representations*, 2020.

[RG22] Maria Refinetti and Sebastian Goldt. The dynamics of representation learning in shallow, non-linear autoencoders. *International Conference on Machine Learning*, 2022.

[RHW86] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation, parallel distributed processing, explorations in the microstructure of cognition, ed. de rumelhart and j. mcclelland. vol. 1. 1986. *Biometrika*, 1986.

[RMB$^+$18] Akshay Rangamani, Anirbit Mukherjee, Amitabh Basu, Ashish Arora, Tejaswini Ganapathi, Sang Chin, and Trac D Tran. Sparse coding and autoencoders. In *IEEE International Symposium on Information Theory*, 2018.

[RMW14] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, 2014.

[Ros58] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 1958.

[RR08] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in Neural Information Processing Systems*, 2008.

[RSF19] Sundeep Rangan, Philip Schniter, and Alyson K Fletcher. Vector approximate message passing. *IEEE Transactions on Information Theory*, 2019.

[RVE18] Grant M. Rotskoff and Eric Vanden-Eijnden. Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error. *Advances in Neural Information Processing Systems*, 2018.

[San17] Filippo Santambrogio. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 2017.

[SBGG23] Eszter Székely, Lorenzo Bardone, Federica Gerace, and Sebastian Goldt. Learning from higher-order statistics, efficiently: hypothesis tests, random features, and neural networks. *arXiv preprint*, 2023.

[SCS$^+$22] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 2022.

[SESS19] Pedro Savarese, Itay Evron, Daniel Soudry, and Nathan Srebro. How do infinite width bounded norm networks look in function space? *Conference on Learning Theory*, 2019.

[SGd$^+$19] Stefano Spigler, Mario Geiger, Stéphane d'Ascoli, Levent Sagun, Giulio Biroli, and Matthieu Wyart. A jamming transition from under- to over-parametrization affects generalization in deep learning. *Journal of Physics A: Mathematical and Theoretical*, 2019.

[Sha48] C. E. Shannon. Mathematical theory of communication. *The Bell System Technical Journal*, 1948.

[Sha59] C. E. Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE National Convention Record*, 1959.

[SHM$^+$16] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 2016.

[SHN⁺18] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 2018.

[SKHM23] Aleksandr Shevchenko, Kevin Kögler, Hamed Hassani, and Marco Mondelli. Fundamental limits of two-layer autoencoders, and achieving them with gradient methods. *International Conference on Machine Learning*, 2023.

[SKM22] Alexander Shevchenko, Vyacheslav Kungurtsev, and Marco Mondelli. Mean-field analysis of piecewise linear solutions for wide ReLU networks. *Journal of Machine Learning Research*, 2022.

[SKZ⁺23] James B Simon, Maksis Knutins, Liu Ziyin, Daniel Geisz, Abraham J Fetterman, and Joshua Albrecht. On the stepwise nature of self-supervised learning. *International Conference on Machine Learning*, 2023.

[SM20] Alexander Shevchenko and Marco Mondelli. Landscape connectivity and dropout stability of SGD solutions for over-parameterized neural networks. *International Conference on Machine Learning*, 2020.

[SNW23] Taiji Suzuki, Atsushi Nitanda, and Denny Wu. Uniform-in-time propagation of chaos for the mean-field gradient langevin dynamics. *International Conference on Learning Representations*, 2023.

[SPD⁺22] Justin Sahs, Ryan Pyle, Aneel Damaraju, Josue Ortega Caro, Onur Tavaslioglu, Andy Lu, and Ankit Patel. Shallow univariate ReLU networks as splines: initialization, loss surface, hessian, & gradient flow dynamics. *Frontiers in Artificial Intelligence*, 2022.

[SRF16] Philip Schniter, Sundeep Rangan, and Alyson K Fletcher. Vector approximate message passing for the generalized linear model. *Asilomar Conference on Signals, Systems and Computers*, 2016.

[SS16] Itay Safran and Ohad Shamir. On the quality of the initial basin in overspecified neural networks. *International Conference on Machine Learning*, 2016.

[SS18a] Itay Safran and Ohad Shamir. Spurious local minima are common in two-layer ReLU neural networks. *International Conference on Machine Learning*, 2018.

[SS18b] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks. *arXiv preprint*, 2018.

[SS19a] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of deep neural networks. *arXiv preprint*, 2019.

[SS19b] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: A central limit theorem. *Stochastic Processes and their Applications*, 2019.

[SS20] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: A law of large numbers. *SIAM Journal on Applied Mathematics*, 2020.

[SZ15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*, 2015.

[Szn91] Alain-Sol Sznitman. Topics in propagation of chaos. *Ecole d'été de probabilités de Saint-Flour XIX—1989*, 1991.

[Tak19] Keigo Takeuchi. Rigorous dynamics of expectation-propagation-based signal recovery from unitarily invariant measurements. *IEEE Transactions on Information Theory*, 2019.

[TBL18] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *Workshop on Bayesian Deep Learning at Advances in Neural Information Processing Systems*, 2018.

[TCVS13] Antonia M Tulino, Giuseppe Caire, Sergio Verdú, and Shlomo Shamai. Support recovery with sparsely sampled free random matrices. *IEEE Transactions on Information Theory*, 2013.

[TSCH17] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *International Conference on Learning Representations*, 2017.

[TSHM22] Lucas Theis, Tim Salimans, Matthew D Hoffman, and Fabian Mentzer. Lossy compression with gaussian diffusion. *arXiv preprint*, 2022.

[TUK06] Koujin Takeda, Shinsuke Uda, and Yoshiyuki Kabashima. Analysis of CDMA systems that are characterized by eigenvalue spectrum. *Europhysics Letters*, 2006.

[VBB19] Luca Venturi, Afonso Bandeira, and Joan Bruna. Spurious valleys in two-layer neural network optimization landscapes. *Journal of Machine Learning Research*, 2019.

[VDOV$^+$17] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 2017.

[Ver18] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge University Press, 2018.

[Vil09] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[Vis00] George Visick. A quantitative version of the observation that the hadamard product is a principal submatrix of the kronecker product. *Linear Algebra and Its Applications*, 2000.

[VKM22] Ramji Venkataramanan, Kevin Kögler, and Marco Mondelli. Estimation in rotationally invariant generalized linear models via approximate message passing. *International Conference on Machine Learning*, 2022.

[VLBM08] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. *International Conference on Machine Learning*, 2008.

[VLS11] Ulrike Von Luxburg and Bernhard Schölkopf. Statistical learning theory: Models, concepts, and results. *Handbook of the History of Logic*, 2011.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[WB21] Aaron B. Wagner and Johannes Ballé. Neural networks optimally compress the sawbridge. *Data Compression Conference*, 2021.

[Win12] Andreas Winkelbauer. Moments and absolute moments of the normal distribution. *arXiv preprint*, 2012.

[WLLM18] Colin Wei, Jason Lee, Qiang Liu, and Tengyu Ma. On the margin theory of feedforward neural networks. *arXiv preprint*, 2018.

[WMM10] Martin J Wainwright, Elitza Maneva, and Emin Martinian. Lossy source compression using low-density generator matrix codes: Analysis and algorithms. *IEEE Transactions on Information Theory*, 2010.

[Wri15] Stephen J Wright. Coordinate descent algorithms. *Mathematical Programming*, 2015.

[WTS+19] Francis Williams, Matthew Trager, Claudio Silva, Daniele Panozzo, Denis Zorin, and Joan Bruna. Gradient dynamics of shallow univariate ReLU networks. *Advances in Neural Information Processing Systems*, 2019.

[WZBG21] Jingfeng Wu, Difan Zou, Vladimir Braverman, and Quanquan Gu. Direction matters: On the implicit bias of stochastic gradient descent with moderate learning rate. *International Conference on Learning Representations*, 2021.

[WZF24] Tianhao Wang, Xinyi Zhong, and Zhou Fan. Universality of approximate message passing algorithms and tensor networks. *Annals of Applied Probability*, 2024.

[YFW+03] Jonathan S Yedidia, William T Freeman, Yair Weiss, et al. Understanding belief propagation and its generalizations. *Exploring Artificial Intelligence in the New Millennium*, 2003.

[YLZ+19] Penghang Yin, Jiancheng Lyu, Shuai Zhang, Stanley Osher, Yingyong Qi, and Jack Xin. Understanding straight-through estimator in training activation quantized neural nets. *International Conference on Learning Representations*, 2019.

[YM21a] Yibo Yang and Stephan Mandt. Lower bounding rate-distortion from samples. *Workshop on Neural Compression: From Information Theory to Applications at International Conference on Learning Representations*, 2021.

[YM21b] Yibo Yang and Stephan Mandt. Towards empirical sandwich bounds on the rate-distortion function. *International Conference on Learning Representations*, 2021.

[YM23] Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. *Advances in Neural Information Processing Systems*, 2023.

[YMT22] Yibo Yang, Stephan Mandt, and Lucas Theis. An introduction to neural data compression. *arXiv preprint*, 2022.

[YSJ18] Chulhee Yun, Suvrit Sra, and Ali Jadbabaie. A critical view of global optimality in deep learning. *International Conference on Learning Representations*, 2018.

[ZCZG20] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Gradient descent optimizes over-parameterized deep relu networks. *Machine Learning*, 2020.

[ZXLM20] Yaoyu Zhang, Zhi-Qin John Xu, Tao Luo, and Zheng Ma. A type of generalization error induced by initialization in deep neural networks. *Mathematical and Scientific Machine Learning*, 2020.

APPENDIX A

Appendix for Chapter 3

A.1 Proof of Theorem 1

A.1.1 Part (A)

Given $\boldsymbol{\theta} = (a, \boldsymbol{w}) \in \mathbb{R}^D$, let $\sigma_\star(\boldsymbol{x}, \boldsymbol{\theta}) = a\sigma(\boldsymbol{x}, \boldsymbol{w})$. Given $\rho \in \mathscr{P}(\mathbb{R}^D)$, we define the limit loss as

$$\bar{L}(\rho) = \mathbb{E}\left\{\left(y - \int \sigma_\star(\boldsymbol{x}, \boldsymbol{\theta})\rho(\mathrm{d}\boldsymbol{\theta})\right)^2\right\}, \tag{A.1}$$

where the expectation is taken over $(\boldsymbol{x}, y)$. For $i \in [N]$ and $t \geq 0$, we consider the following nonlinear dynamics:

$$\frac{\mathrm{d}}{\mathrm{d}t}\bar{\boldsymbol{\theta}}_i^t = 2\xi(t)\int \mathbb{E}\left\{\nabla\sigma_\star(\boldsymbol{x}, \bar{\boldsymbol{\theta}}_i^t)\left(y - \sigma_\star(\boldsymbol{x}, \boldsymbol{\theta}')\right)\right\}\rho_t(\mathrm{d}\boldsymbol{\theta}'), \tag{A.2}$$

where ∇ denotes the gradient with respect to $\bar{\boldsymbol{\theta}}_i^t$ and $\bar{\boldsymbol{\theta}}_i^t \sim \rho_t$. We initialize (A.2) with $\{\bar{\boldsymbol{\theta}}_i^0\}_{i=1}^N \overset{\text{i.i.d.}}{\sim} \rho_0$.

In [MMM19], it is considered the two-layer neural network (3.2) with N neurons and bounded activation function σ, and it is studied the evolution under the SGD algorithm (3.4) of the parameters $\boldsymbol{\theta}^k$. In particular, it is shown that, under suitable assumptions, *(i)* the solution of (A.2) exists and it is unique, *(ii)* the N i.i.d. ideal particles $\{\bar{\boldsymbol{\theta}}_i^t\}_{i=1}^N$ are close to the parameters $\boldsymbol{\theta}^k$ obtained after k steps of SGD with step size α, with $t = k\alpha$, and *(iii)* the loss $L_N(\boldsymbol{\theta}^k)$ concentrates to the limit loss $\bar{L}(\rho_t)$, where ρ_t is the law of $\bar{\boldsymbol{\theta}}_i^t$.

Let us now provide the proof of Theorem 1, part (A).

Proof of Theorem 1, part (A). Without loss of generality, we can assume that $\boldsymbol{\theta}_{\mathrm{S}}^k$ contains the first $|\mathcal{A}|$ elements of $\boldsymbol{\theta}^k$, i.e., $\boldsymbol{\theta}_{\mathrm{S}}^k = (\boldsymbol{\theta}_1^k, \boldsymbol{\theta}_2^k, \ldots, \boldsymbol{\theta}_{|\mathcal{A}|}^k)$. In fact, the subset $\mathcal{A}$ is independent of the SGD algorithm. Thus, by symmetry, the joint distribution of $\{\boldsymbol{\theta}_i^k\}_{i\in\mathcal{A}}$ depends only on $|\mathcal{A}|$ (and not on the set $\mathcal{A}$ itself). By Definition 3.3.1, we need to show that, with probability at least $1 - e^{-z^2}$,

$$\sup_{k\in[T/\alpha]} |L_N(\boldsymbol{\theta}^k) - L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}^k)| \leq Ke^{KT^3}\left(\frac{\sqrt{\log|\mathcal{A}|} + z}{\sqrt{|\mathcal{A}|}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right). \tag{A.3}$$

Let $\bar{\boldsymbol{\theta}}^{k\alpha} = (\bar{\boldsymbol{\theta}}_1^{k\alpha}, \ldots, \bar{\boldsymbol{\theta}}_N^{k\alpha})$ be the solution of the nonlinear dynamics (A.2) at time $k\alpha$, with $\bar{\boldsymbol{\theta}}_i^{k\alpha} \sim \rho_{k\alpha}$. By triangle inequality, we have that

$$\begin{aligned}|L_N(\boldsymbol{\theta}^k) - L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}^k)| &\le |L_N(\boldsymbol{\theta}^k) - \bar{L}(\rho_{k\alpha})| + |L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}^k) - \bar{L}(\rho_{k\alpha})| \\ &\le |L_N(\boldsymbol{\theta}^k) - \bar{L}(\rho_{k\alpha})| + |L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}^k) - L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\mathrm{S}}^{k\alpha})| + |L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\mathrm{S}}^{k\alpha}) - \bar{L}(\rho_{k\alpha})|,\end{aligned} \tag{A.4}$$

where $\bar{L}$ is defined in (A.1) and $\bar{\boldsymbol{\theta}}_{\mathrm{S}}^{k\alpha} = (\bar{\boldsymbol{\theta}}_1^{k\alpha}, \bar{\boldsymbol{\theta}}_2^{k\alpha}, \ldots, \bar{\boldsymbol{\theta}}_{|\mathcal{A}|}^{k\alpha})$ denotes the vector containing the first $|\mathcal{A}|$ elements of $\bar{\boldsymbol{\theta}}^{k\alpha}$.

Let us consider the first term in the RHS of (A.4). Note that, without loss of generality, we can assume that $\alpha \le 1/(C(D + \log N + z^2)e^{CT^3})$, for some constant C depending only on the constants K_i of the assumptions **(A1)**-**(A4)**. Let us explain why this is the case. If $\alpha > 1/(C(D + \log N + z^2)e^{CT^3})$, then the RHS of (A.3) is lower bounded by a constant depending only on K_i. Furthermore, y and σ are bounded, and by Proposition 8 of [MMM19], we have that, with probability at least $1 - e^{-z^2}$,

$$\sup_{k \in [T/\alpha]} \max_{i \in [N]} |a_i^k| \le C_3(1 + T). \tag{A.5}$$

Thus, if $\alpha > 1/(C(D + \log N + z^2)e^{CT^3})$, then the result is trivially true. Consequently, we can apply Theorem 1 of [MMM19] and we have that, with probability at least $1 - e^{-z^2}$,

$$\sup_{k \in [T/\alpha]} |L_N(\boldsymbol{\theta}^k) - \bar{L}(\rho_{k\alpha})| \le C_1 e^{C_1 T^3} \left(\frac{\sqrt{\log N} + z}{\sqrt{N}} + \sqrt{\alpha}\Big(\sqrt{D + \log N} + z\Big) \right), \tag{A.6}$$

where C_1 depends only on K_i. In what follows, the C_i are constants that depend only on K_i.

Let us now consider the second term in the RHS of (A.4). After some manipulations, we have that

$$\begin{aligned}|L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}^k) - L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\mathrm{S}}^{k\alpha})| &\le 2 \max_{i \in \mathcal{A}} \left| a_i^k \mathbb{E}\left\{ y\sigma(\boldsymbol{x}, \boldsymbol{w}_i^k) \right\} - \bar{a}_i^{k\alpha} \mathbb{E}\left\{ y\sigma(\boldsymbol{x}, \bar{\boldsymbol{w}}_i^{k\alpha}) \right\} \right| \\ &\quad + \max_{i,j \in \mathcal{A}} \left| a_i^k a_j^k \mathbb{E}\{\sigma(\boldsymbol{x}, \boldsymbol{w}_i^k)\sigma(\boldsymbol{x}, \boldsymbol{w}_j^k)\} - \bar{a}_i^{k\alpha} \bar{a}_j^{k\alpha} \mathbb{E}\{\sigma(\boldsymbol{x}, \bar{\boldsymbol{w}}_i^{k\alpha})\sigma(\boldsymbol{x}, \bar{\boldsymbol{w}}_j^{k\alpha})\} \right| \\ &\le C_2 \left(\max_{i \in \mathcal{A}} \Big(1 + \max(|a_i^k|, |\bar{a}_i^{k\alpha}|)\Big) \right)^2 \max_{i \in \mathcal{A}} \|\boldsymbol{\theta}_i^k - \bar{\boldsymbol{\theta}}_i^{k\alpha}\|_2 \\ &\le C_2 \left(\max_{i \in [N]} \Big(1 + \max(|a_i^k|, |\bar{a}_i^{k\alpha}|)\Big) \right)^2 \max_{i \in [N]} \|\boldsymbol{\theta}_i^k - \bar{\boldsymbol{\theta}}_i^{k\alpha}\|_2,\end{aligned} \tag{A.7}$$

where $\boldsymbol{\theta}_i^k = (a_i^k, \boldsymbol{w}_i^k)$, $\bar{\boldsymbol{\theta}}_i^{k\alpha} = (\bar{a}_i^{k\alpha}, \bar{\boldsymbol{w}}_i^{k\alpha})$, and in the second inequality we use that y, σ and the gradient of σ are bounded. By using Lemma 7 of [MMM19], we have that

$$\sup_{t \in [0,T]} \max_{i \in [N]} |\bar{a}_i^t| \le C_3(1 + T). \tag{A.8}$$

Furthermore, by using Propositions 6-7-8 of [MMM19], we have that, with probability at least $1 - e^{-z^2}$,

$$\sup_{k \in [T/\alpha]} \max_{i \in [N]} \|\boldsymbol{\theta}_i^k - \bar{\boldsymbol{\theta}}_i^{k\alpha}\|_2 \le C_4 e^{C_4 T^3} \left(\frac{\sqrt{\log N} + z}{\sqrt{N}} + \sqrt{\alpha}\Big(\sqrt{D + \log N} + z\Big) \right). \tag{A.9}$$

As a result, by combining (A.5), (A.8) and (A.7), we conclude that, with probability at least $1-e^{-z^2}$,

$$\sup_{k\in[T/\alpha]} |L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}^k) - L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha})| \le C_5 e^{C_5 T^3}\left(\frac{\sqrt{\log N}+z}{\sqrt{N}} + \sqrt{\alpha}\Big(\sqrt{D+\log N}+z\Big)\right). \tag{A.10}$$

Finally, let us consider the third term in the RHS of (A.4). By triangle inequality, we have that

$$|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha}) - \bar{L}(\rho_{k\alpha})| \le \left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha}) - \mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha})\right\}\right| + \left|\mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha})\right\} - \bar{L}(\rho_{k\alpha})\right|, \tag{A.11}$$

where the notation $\mathbb{E}_{\rho_0}$ emphasizes that the expectation is taken with respect to $\bar{\boldsymbol{\theta}}_i^0 \sim \rho_0$. Recall that $\bar{L}$ is defined in (A.1) and that

$$L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}) = \mathbb{E}_{(\boldsymbol{x},y)}\left\{\left(y - \frac{1}{|\mathcal{A}|}\sum_{i=1}^{|\mathcal{A}|}\sigma_\star(\boldsymbol{x},\boldsymbol{\theta}_i)\right)^2\right\}, \tag{A.12}$$

where the notation $\mathbb{E}_{(\boldsymbol{x},y)}$ emphasizes that the expectation is taken with respect to $(\boldsymbol{x},y)\sim\mathbb{P}$. Furthermore, note that $\{\bar{\boldsymbol{\theta}}_i^{k\alpha}\}_{i=1}^{|\mathcal{A}|} \overset{\text{i.i.d.}}{\sim} \rho_{k\alpha}$. Thus, after some manipulations, we can rewrite the second term in the RHS of (A.11) as

$$\begin{aligned}&\left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha}) - \mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha})\right\}\right| \\ &= \frac{1}{|\mathcal{A}|}\left|\int \mathbb{E}_{(\boldsymbol{x},y)}\left\{\Big(\sigma_\star(\boldsymbol{x},\boldsymbol{\theta})\Big)^2\right\}\rho_{k\alpha}(\mathrm{d}\boldsymbol{\theta}) - \int \mathbb{E}_{(\boldsymbol{x},y)}\left\{\sigma_\star(\boldsymbol{x},\boldsymbol{\theta}_1)\sigma_\star(\boldsymbol{x},\boldsymbol{\theta}_2)\right\}\rho_{k\alpha}(\mathrm{d}\boldsymbol{\theta}_1)\rho_{k\alpha}(\mathrm{d}\boldsymbol{\theta}_2)\right|.\end{aligned} \tag{A.13}$$

As σ is bounded by assumption **(A3)** and $\sup_{k\in[T/\alpha]}\max_{i\in[N]}|\bar{a}_i^{k\alpha}|$ is bounded by (A.8), we deduce that

$$\sup_{k\in[T/\alpha]}\left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha}) - \mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^{k\alpha})\right\}\right| \le \frac{C_6\,(1+T)^2}{|\mathcal{A}|}. \tag{A.14}$$

Let $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ be two parameters that differ only in one component, i.e., $\boldsymbol{\theta} = (\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_i,\ldots,\boldsymbol{\theta}_{|\mathcal{A}|})$ and $\boldsymbol{\theta}' = (\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_i',\ldots,\boldsymbol{\theta}_{|\mathcal{A}|})$, and such that $\max_{i\in|\mathcal{A}|}|a_i| \le C(1+T)$ and $\max_{i\in|\mathcal{A}|}|a_i'| \le C(1+T)$. Then,

$$\left|L_{|\mathcal{A}|}(\boldsymbol{\theta}) - L_{|\mathcal{A}|}(\boldsymbol{\theta}')\right| \le \frac{C_7\,(1+T)^2}{|\mathcal{A}|}. \tag{A.15}$$

As $\max_{i\in[N]}|\bar{a}_i^t|$ is bounded by (A.8), by applying McDiarmid's inequality, we obtain that

$$\mathbb{P}\left(\left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^t) - \mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^t)\right\}\right| > \delta\right) \le \exp\left(-\frac{|\mathcal{A}|\delta^2}{C_8(1+T)^4}\right). \tag{A.16}$$

Furthermore, we have that

$$\begin{aligned}\left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^t) - L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^s)\right| &\le C_9\left(\max_{i\in[N]}\Big(1+\max(|\bar{a}_i^t|,|\bar{a}_i^s|)\Big)\right)^2\max_{i\in[N]}\|\bar{\boldsymbol{\theta}}_i^t - \bar{\boldsymbol{\theta}}_i^s\|_2 \\ &\le C_{10}(1+T)^4|t-s|,\end{aligned} \tag{A.17}$$

where in the first inequality we use passages similar to those of (A.7), and in the second inequality we use (A.8) and Lemma 9 of [MMM19]. Consequently,

$$\left||L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^t) - \mathbb{E}_{\rho_0}\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^t)\}| - |L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^s) - \mathbb{E}_{\rho_0}\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}^s)\}|\right| \le C_{11}(1+T)^4|t-s|. \tag{A.18}$$

By taking a union bound over $s \in [T/\nu]$ and bounding the difference between time in the interval grid, we deduce that

$$\mathbb{P}\left(\sup_{t\in[0,T]}\left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}^t_{\mathrm{S}}) - \mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}^t_{\mathrm{S}})\right\}\right| \geq \delta + C_{11}(1+T)^4\nu\right) \leq \frac{T}{\nu}\exp\left(-\frac{|\mathcal{A}|\delta^2}{C_8(1+T)^4}\right). \tag{A.19}$$

Pick $\nu = 1/\sqrt{|\mathcal{A}|}$ and $\delta = C_8(1+T)^2(\sqrt{\log(|\mathcal{A}|T)}+z)/\sqrt{|\mathcal{A}|}$. Thus, with probability at least $1-e^{-z^2}$, we have that

$$\sup_{k\in[T/\alpha]}\left|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}^T_{\mathrm{S}}) - \mathbb{E}_{\rho_0}\left\{L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}^T_{\mathrm{S}})\right\}\right| \leq C_{12}\,(1+T)^3\frac{\sqrt{\log|\mathcal{A}|}+z}{\sqrt{|\mathcal{A}|}}. \tag{A.20}$$

By combining (A.14) and (A.20), we conclude that, with probability at least $1-e^{-z^2}$,

$$\sup_{k\in[T/\alpha]}|L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}^T_{\mathrm{S}}) - \bar{L}(\rho_T)| \leq C_{13}\,(1+T)^3\frac{\sqrt{\log|\mathcal{A}|}+z}{\sqrt{|\mathcal{A}|}}. \tag{A.21}$$

Finally, by combining (A.4), (A.6), (A.10) and (A.21), the result readily follows. □

A.1.2 Part (B)

The proof of part (B) is obtained by combining part (A) with the following lemma.

Lemma A.1.1 (Dropout stability implies connectivity – two-layer)**.** *Consider a two-layer neural network with N neurons, as in* (3.2)*. Given $\mathcal{A} = [N/2]$, let $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ be ε-dropout stable as in Definition 3.3.1. Then, $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are ε-connected as in Definition 3.3.2. Furthermore, the path connecting $\boldsymbol{\theta}$ with $\boldsymbol{\theta}'$ consists of 7 line segments.*

Proof of Lemma A.1.1. Let $\boldsymbol{\theta} = ((a_1,\boldsymbol{w}_1),(a_2,\boldsymbol{w}_2),\ldots,(a_N,\boldsymbol{w}_N))$ and

$$\boldsymbol{\theta}' = ((a'_1,\boldsymbol{w}'_1),(a'_2,\boldsymbol{w}'_2),\ldots,(a'_N,\boldsymbol{w}'_N)).$$

For the moment, assume that N is even. Consider the piecewise linear path in parameter space that connects $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$ via the following intermediate points:

$$\begin{aligned}
\boldsymbol{\theta}_1 &= ((2a_1,\boldsymbol{w}_1),(2a_2,\boldsymbol{w}_2),\ldots,(2a_{N/2},\boldsymbol{w}_{N/2}),(0,\boldsymbol{w}_{N/2+1}),(0,\boldsymbol{w}_{N/2+2}),\ldots,(0,\boldsymbol{w}_N)),\\
\boldsymbol{\theta}_2 &= ((2a_1,\boldsymbol{w}_1),(2a_2,\boldsymbol{w}_2),\ldots,(2a_{N/2},\boldsymbol{w}_{N/2}),(0,\boldsymbol{w}'_1),(0,\boldsymbol{w}'_2),\ldots,(0,\boldsymbol{w}'_{N/2})),\\
\boldsymbol{\theta}_3 &= ((0,\boldsymbol{w}_1),(0,\boldsymbol{w}_2),\ldots,(0,\boldsymbol{w}_{N/2}),(2a'_1,\boldsymbol{w}'_1),(2a'_2,\boldsymbol{w}'_2),\ldots,(2a'_{N/2},\boldsymbol{w}'_{N/2})),\\
\boldsymbol{\theta}_4 &= ((0,\boldsymbol{w}'_1),(0,\boldsymbol{w}'_2),\ldots,(0,\boldsymbol{w}'_{N/2}),(2a'_1,\boldsymbol{w}'_1),(2a'_2,\boldsymbol{w}'_2),\ldots,(2a'_{N/2},\boldsymbol{w}'_{N/2})),\\
\boldsymbol{\theta}_5 &= ((2a'_1,\boldsymbol{w}'_1),(2a'_2,\boldsymbol{w}'_2),\ldots,(2a'_{N/2},\boldsymbol{w}'_{N/2}),(0,\boldsymbol{w}'_1),(0,\boldsymbol{w}'_2),\ldots,(0,\boldsymbol{w}'_{N/2})),\\
\boldsymbol{\theta}_6 &= ((2a'_1,\boldsymbol{w}'_1),(2a'_2,\boldsymbol{w}'_2),\ldots,(2a'_{N/2},\boldsymbol{w}'_{N/2}),(0,\boldsymbol{w}'_{N/2+1}),(0,\boldsymbol{w}'_{N/2+2}),\ldots,(0,\boldsymbol{w}'_N)).
\end{aligned} \tag{A.22}$$

We will now show that the loss along this path is upper bounded by $\max(L_N(\boldsymbol{\theta}),L_N(\boldsymbol{\theta}'))+\varepsilon$.

Consider the path that connects $\boldsymbol{\theta}$ to $\boldsymbol{\theta}_1$. As $\boldsymbol{\theta}$ is ε-dropout stable, we have that $L_N(\boldsymbol{\theta}_1) \leq L_N(\boldsymbol{\theta})+\varepsilon$. As the loss is convex in the weights of the last layer, the loss along this path is upper bounded by $L_N(\boldsymbol{\theta})+\varepsilon$. Similarly, the loss along the path that connects $\boldsymbol{\theta}_6$ to $\boldsymbol{\theta}'$ is upper bounded by $L_N(\boldsymbol{\theta}')+\varepsilon$.

Consider the path that connects $\boldsymbol{\theta}_1$ to $\boldsymbol{\theta}_2$. Here, we change $\boldsymbol{w}$'s only when the corresponding a's are 0. Thus, the loss does not change along the path. Similarly, the loss does not change along the path that connects $\boldsymbol{\theta}_3$ to $\boldsymbol{\theta}_4$ and $\boldsymbol{\theta}_5$ to $\boldsymbol{\theta}_6$.

Consider the path that connects $\boldsymbol{\theta}_2$ to $\boldsymbol{\theta}_3$. Note that $L_N(\boldsymbol{\theta}_3) = L_N(\boldsymbol{\theta}_5)$. As the loss is convex in the weights of the last layer, the loss along this path is upper bounded by $\max(L_N(\boldsymbol{\theta}), L_N(\boldsymbol{\theta}')) + \varepsilon$.

Finally, consider the path that connects $\boldsymbol{\theta}_4$ to $\boldsymbol{\theta}_5$. Here, we are interpolating between two equal subnetworks. Thus, the loss along this path does not change. This concludes the proof for even N.

If N is odd, a similar argument can be carried out. The differences are that *(i)* the $\lceil N/2 \rceil$-th parameter of $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$ and $\boldsymbol{\theta}_3$ is $(0, \boldsymbol{w}_{N/2})$ and the $\lceil N/2 \rceil$-th parameter of $\boldsymbol{\theta}_4$, $\boldsymbol{\theta}_5$ and $\boldsymbol{\theta}_6$ is $(0, \boldsymbol{w}'_{N/2})$, and *(ii)* the constant 2 in front of the a_i is replaced by $N/\lfloor N/2 \rfloor$. □

A.2 Extension to Unbounded Activation – Statement and Proof

We modify the assumptions **(A2)**, **(A3)** and **(A4)** of Section 3.3.2 as follows:

(A2') The feature vectors $\boldsymbol{x}$ and the response variables y are bounded by K_2, and the gradient $\nabla_{\boldsymbol{w}}\sigma(\boldsymbol{x}, \boldsymbol{w})$ is K_2 sub-gaussian when $\boldsymbol{x} \sim \mathbb{P}$.

(A3') The activation function σ is differentiable, with gradient bounded by K_3 and K_3-Lipschitz.

(A4') The initialization ρ_0 is supported on $\|\boldsymbol{\theta}_i^0\|_2 \le K_4$.

We are now ready to present our results for unbounded activations in the two-layer setting.

Theorem 10 (Two-layer, unbounded activation)**.** *Assume that conditions **(A1)**, **(A2')**, **(A3')** and **(A4')** hold, and fix $T \ge 1$. Let $\boldsymbol{\theta}^k$ be obtained by running k steps of the SGD algorithm* (3.4) *with data $\{(\boldsymbol{x}_j, y_j)\}_{j=0}^k \overset{\text{i.i.d.}}{\sim} \mathbb{P}$ and initialization ρ_0. Assume further that the loss at each step of SGD is uniformly bounded, i.e., $\max_{j\in\{0,\dots,k\}} |y_j - \hat{\boldsymbol{y}}_N(\boldsymbol{x}_j, \boldsymbol{\theta}^j)| \le K_5$. Then, the results of Theorem 1 hold, with*

$$\begin{aligned}\varepsilon_{\mathrm{D}} &= K(T)\left(\frac{\sqrt{\log|\mathcal{A}|}+z}{\sqrt{|\mathcal{A}|}} + \sqrt{\alpha}\Big(\sqrt{D+\log N}+z\Big)\right),\\ \varepsilon_{\mathrm{C}} &= K(\max(T,T'))\left(\frac{\sqrt{\log N}+z}{\sqrt{N}} + \sqrt{\alpha}\Big(\sqrt{D+\log N}+z\Big)\right).\end{aligned} \tag{A.23}$$

where the constant $K(T)$ depends on T and on the constants K_i of the assumptions.

To prove the result, we crucially rely on the following bound on the norm of the parameters evolved via SGD.

Lemma A.2.1 (Bound on norm of SGD parameters)**.** *Under the assumptions of Theorem 10, we have that*

$$\sup_{s\in[T/\alpha]} \max_{i\in[N]} \|\boldsymbol{\theta}_i^s\|_2 \le K e^{KT}, \tag{A.24}$$

where the constant K depends only on the constants K_i of the assumptions.

Proof of Lemma A.2.1. The SGD update at step $j+1$ gives:

$$\begin{aligned} a_i^{j+1} &= a_i^j + 2\alpha\,\xi(j\alpha)\cdot(y_j - f_N(\boldsymbol{x}_j,\boldsymbol{\theta}^j))\cdot\sigma(\boldsymbol{x}_j,\boldsymbol{w}_i^j),\\ \boldsymbol{w}_i^{j+1} &= \boldsymbol{w}_i^j + 2\alpha\,\xi(j\alpha)\cdot(y_j - f_N(\boldsymbol{x}_j,\boldsymbol{\theta}^j))\cdot a_i^j\nabla_{\boldsymbol{w}_i}\sigma(\boldsymbol{x}_j,\boldsymbol{w}_i^j). \end{aligned} \tag{A.25}$$

We bound the absolute value of the increment $|a_i^{j+1}-a_i^j|$ as

$$\begin{aligned} |a_i^{j+1}-a_i^j| &\le 2\alpha\xi(j\alpha)\cdot|y_j - f_N(\boldsymbol{x}_j,\boldsymbol{\theta}^j)|\cdot|\sigma(\boldsymbol{x}_j,\boldsymbol{w}_i^j)|\\ &\overset{(a)}{\le} \alpha C_1|\sigma(\boldsymbol{x}_j,\boldsymbol{w}_i^j)|\\ &\overset{(b)}{\le} \alpha C_2(\|\boldsymbol{w}_i^j\|_2+1), \end{aligned} \tag{A.26}$$

where the constant C_i depends only on K_i, in (a) we use that ξ is bounded by K_1 and $|y_j - f_N(\boldsymbol{x}_j,\boldsymbol{\theta}^j)| \le K_5$, in (b) we use that $\|\sigma\|_{\mathrm{Lip}} \le K_2$ and $\|\boldsymbol{x}_j\|_2 \le K_2$. Similarly, we bound the absolute value of the increments $\|\boldsymbol{w}_i^{j+1}-\boldsymbol{w}_i^j\|_2$ as

$$\|\boldsymbol{w}_i^{j+1}-\boldsymbol{w}_i^j\|_2 \le \alpha C_3|a_i^j|. \tag{A.27}$$

By combining (A.26) and (A.27), we get

$$\|\boldsymbol{\theta}_i^{j+1}-\boldsymbol{\theta}_i^j\|_2 \le \|\boldsymbol{w}_i^{j+1}-\boldsymbol{w}_i^j\|_2 + |a_i^{j+1}-a_i^j| \le \alpha C_4(\|\boldsymbol{\theta}_i^j\|_2+1). \tag{A.28}$$

By triangle inequality, we also obtain that

$$\|\boldsymbol{\theta}_i^s\|_2 \le \sum_{j=0}^{s-1}\|\boldsymbol{\theta}_i^{j+1}-\boldsymbol{\theta}_i^j\|_2 + \|\boldsymbol{\theta}_i^0\|_2. \tag{A.29}$$

As $\|\boldsymbol{\theta}_i^0\|_2$ is bounded, by combining (A.28) and (A.29), we have that

$$\|\boldsymbol{\theta}_i^s\|_2 \le C_5 + C_5\,s\alpha + C_5\alpha\sum_{j=0}^{s-1}\|\boldsymbol{\theta}_i^j\|_2. \tag{A.30}$$

By using a discrete version of Gronwall's inequality, the result follows. □

Finally, let us present the proof of Theorem 10.

Proof of Theorem 10. Since the activation function σ satisfies assumption **(A3')**, we can construct $\tilde{\sigma}:\mathbb{R}^d\times\mathbb{R}^{D-1}\to\mathbb{R}$ that satisfies the following two properties:

(i) $\tilde{\sigma}(\boldsymbol{x},\boldsymbol{w})$ coincides with $\sigma(\boldsymbol{x},\boldsymbol{w})$ for $\|\boldsymbol{x}\|_2 \le K_2$ and $\|\boldsymbol{w}\|_2 \le Ke^{KT}$, where K_2 is the constant of assumption **(A2')** and Ke^{KT} is the bound of Lemma A.2.1;

(ii) $\tilde{\sigma}(\boldsymbol{x},\boldsymbol{w})$ is bounded, differentiable, with bounded and Lipschitz continuous gradient.

Recall that $\boldsymbol{\theta}^k$ is obtained by running k steps of the SGD algorithm (3.4) with initial condition $\boldsymbol{\theta}^0$, data $\{\boldsymbol{x}_j,y_j\}_{j=0}^k$ and activation function σ. Let $\tilde{\boldsymbol{\theta}}^k$ be obtained by running k steps of SGD with initial condition $\boldsymbol{\theta}^0$, data $\{\boldsymbol{x}_j,y_j\}_{j=0}^k$ and activation function $\tilde{\sigma}$. By combining Lemma A.2.1, assumption **(A2')** and property **(i)** of $\tilde{\sigma}$, we immediately deduce that

$$\boldsymbol{\theta}^k = \tilde{\boldsymbol{\theta}}^k. \tag{A.31}$$

Furthermore, we have that

$$\mathbb{E}\left\{\left(y-\frac{1}{N}\sum_{i=1}^{N} a_i\sigma(\boldsymbol{x},\boldsymbol{w}_i))\right)^2\right\} = \mathbb{E}\left\{\left(y-\frac{1}{N}\sum_{i=1}^{N} a_i\tilde{\sigma}(\boldsymbol{x},\boldsymbol{w}_i))\right)^2\right\}, \tag{A.32}$$

namely the loss of $\boldsymbol{\theta}^k$ computed with respect to the activation function σ is the same as the loss of $\boldsymbol{\theta}^k$ computed with respect to the activation function $\tilde{\sigma}$.

Note that $\|\tilde{\sigma}\|_\infty \le C_1(T)$ for some $C_1(T)$ that depends on T and on the constants K_i of the assumptions. Thus, $\tilde{\sigma}$ satisfies assumptions **(A2)** and **(A3)**, with K_3 depending on time T of the evolution. Consequently, by Theorem 1, with probability at least $1-e^{-z^2}$, for all $k \in [T/\alpha]$, $\tilde{\boldsymbol{\theta}}^k$ is ε_{D}-dropout stable, with

$$\varepsilon_{\mathrm{D}} = K(T)\left(\sqrt{\frac{\log N}{N}} + \frac{\sqrt{\log|\mathcal{A}|+z}}{\sqrt{|\mathcal{A}|}} + \sqrt{\alpha}\Big(\sqrt{D+\log N}+z\Big)\right). \tag{A.33}$$

By using (A.31) and (A.32), we conclude that, with probability at least $1-e^{-z^2}$, for all $k \in [T/\alpha]$, $\boldsymbol{\theta}^k$ is ε_{D}-dropout stable. Similarly, with probability at least $1-e^{-z^2}$, for all $k' \in [T'/\alpha]$, $(\boldsymbol{\theta}')^{k'}$ is ε_{D}-dropout stable. Thus, by Lemma A.1.1, the proof is complete. □

A.3 Proof of Theorem 2

A.3.1 Part (A)

Let $D=\sum_{i=0}^{L} D_i$ and let ρ be a probability measure over $\mathbb{R}^D \cong \mathbb{R}^{D_0}\times\mathbb{R}^{D_1}\times\cdots\times\mathbb{R}^{D_L}$. For $i\in\{0,\dots,L\}$, we denote by $\rho^{(i)}$ the marginal of ρ over the i-th factor $\mathbb{R}^{D_i}$ of the Cartesian product. For $i\in\{0,\dots,L-1\}$, we denote by $\rho^{(i,i+1)}$ the marginal of ρ over the i-th and the $i+1$-th factors. Furthermore, we denote by $\rho^{(i|i+1)}(\cdot \mid \boldsymbol{\theta}^{(i+1)})$ the conditional distribution of the i-th factor given that the $i+1$-th factor is equal to $\boldsymbol{\theta}^{(i+1)}$.

Given a feature vector $\boldsymbol{x}\in\mathbb{R}^{d_0}$ and a probability measure ρ over $\mathbb{R}^D$, we define

$$\begin{aligned}
\bar{\boldsymbol{z}}^{(2)}(\boldsymbol{x},\rho) &= \int \sigma^{(1)}\left(\sigma^{(0)}\left(\boldsymbol{x},\boldsymbol{\theta}^{(0)}\right),\boldsymbol{\theta}^{(1)}\right)\,\mathrm{d}\rho^{(0,1)}(\boldsymbol{\theta}^{(0)},\boldsymbol{\theta}^{(1)}),\\
\bar{\boldsymbol{z}}^{(\ell)}(\boldsymbol{x},\rho) &= \int \sigma^{(\ell-1)}\left(\bar{\boldsymbol{z}}^{(\ell-1)}(\boldsymbol{x},\rho),\boldsymbol{\theta}^{(\ell-1)}\right)\,\mathrm{d}\rho^{(\ell-1)}(\boldsymbol{\theta}^{(\ell-1)}), \qquad \ell\in\{3,\dots,L-1\},\\
\bar{\boldsymbol{z}}^{(L)}\left(\boldsymbol{x},\rho,\boldsymbol{\theta}^{(L)}\right) &= \int \sigma^{(L-1)}\left(\bar{\boldsymbol{z}}^{(L-1)}(\boldsymbol{x},\rho),\boldsymbol{\theta}^{(L-1)}\right)\,\mathrm{d}\rho^{(L-1|L)}(\boldsymbol{\theta}^{(L-1)}\mid\boldsymbol{\theta}^{(L)}),\\
\bar{\boldsymbol{y}}(\boldsymbol{x},\rho) &= \sigma^{(L+1)}\left(\int \sigma^{(L)}\left(\bar{\boldsymbol{z}}^{(L)}\left(\boldsymbol{x},\rho,\boldsymbol{\theta}^{(L)}\right),\boldsymbol{\theta}^{(L)}\right)\,\mathrm{d}\rho^{(L)}(\boldsymbol{\theta}^{(L)})\right),
\end{aligned} \tag{A.34}$$

where $\sigma^{(\ell)}:\mathbb{R}^{d_\ell}\times\mathbb{R}^{D_\ell}\to\mathbb{R}^{d_{\ell+1}}$, with $\ell\in\{0,\dots,L\}$, and $\sigma^{(L+1)}:\mathbb{R}^{d_{L+1}}\to\mathbb{R}^{d_{L+1}}$. We remark that $\bar{\boldsymbol{z}}^{(L)}$ is defined in terms of the conditional distribution $\rho^{(L-1|L)}$. We also define the limit loss as

$$\bar{L}(\rho) = \mathbb{E}\left\{\|\boldsymbol{y}-\bar{\boldsymbol{y}}(\boldsymbol{x},\rho)\|_2^2\right\}, \tag{A.35}$$

where the expectation is taken over $(\boldsymbol{x},\boldsymbol{y})$. Given a probability measure ρ_0 over $\mathbb{R}^D$ and activation functions $\sigma^{(\ell)}$ ($\ell\in\{0,\dots,L+1\}$), we denote by $\rho^\star_{[0,T]}$ the probability measure over $\mathcal{C}([0,T],\mathbb{R}^D)$ which solves the McKean-Vlasov DNN problem with initial condition ρ_0, according to Definition 4.4 of [AOY19]. We also denote by $\rho^\star_t$ the marginal of $\rho^\star_{[0,T]}$ at time $t \in [0, T]$.

In [AOY19], it is considered a model of neural network with $L + 1 \geq 4$ layers, where each hidden layer contains N neurons. This model can be obtained from (3.10) by setting to one the parameters $\{\boldsymbol{a}^\ell_{i_\ell,i_{\ell+1}}\}_{\ell\in[L-1],i_\ell,i_{\ell+1}\in[N]}$ and $\{\boldsymbol{a}^{(L)}_{i_L}\}_{i_L\in[N]}$, and by applying the bounded activation function $\sigma^{(L+1)}$ to the output $\widehat{\boldsymbol{y}}_N$. Then, it is studied the evolution under the SGD algorithm (3.12) of the parameters $\boldsymbol{\theta}(k)$ of this multilayer neural network. In particular, it is shown that, under suitable assumptions, *(i)* the solution of the McKean-Vlasov DNN problem exists and it is unique, *(ii)* the parameters $\boldsymbol{\theta}(k)$ obtained after k steps of SGD with step size α are close to particles $\bar{\boldsymbol{\theta}}(t)$ at time $t = k\alpha$, whose trajectories are distributed according to $\rho^\star_t$, and *(iii)* the loss $L_N(\boldsymbol{\theta}(k))$ concentrates to the limit loss $\bar{L}(\rho^\star_t)$.

In order to prove Theorem 2, we will use the following bound on the norm of the parameters $\{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}\}_{\ell\in[L-1],i_\ell,i_{\ell+1}\in[N]}$ evolved via SGD.

Lemma A.3.1 (Bound on norm of $\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}$). *Under the assumptions of Theorem 2, we have that*

$$\max_{\ell\in[L-1]} \sup_{s\in[T/\alpha]} \max_{i_\ell,i_{\ell+1}\in[N]} \|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(s)\|_2 \leq K(T, L), \tag{A.36}$$

where the constant K depends only on T, L and on the constants K_i of the assumptions.

Proof. For $\ell \in [L - 1]$, the SGD update at step $j + 1$ gives:

$$\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j+1) = \boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)+2\alpha\xi(j\alpha)N^2 \left(\boldsymbol{y}_j - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_j, \boldsymbol{\theta}(j)\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right), \tag{A.37}$$

where $\mathrm{D}_{\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N \in \mathbb{R}^{d_{L+1}} \times \mathbb{R}^{D_\ell+d_{\ell+1}}$ denotes the Jacobian of $\widehat{\boldsymbol{y}}_N$ with respect to $\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}$.

Recall that by assumptions **(B2)-(B3)** the response variables $\boldsymbol{y}_j$ and the activation $\sigma^{(L)}$ are bounded. Moreover, as the final layer of the network is not trained, i.e., $\boldsymbol{a}^{(L)}_{i_L}(k + 1) = \boldsymbol{a}^{(L)}_{i_L}(k)$ for any k, and $\boldsymbol{a}^{(L)}_{i_L}(0)$ is initialized with a distribution supported on $\|\boldsymbol{a}^{(L)}_{i_L}(0)\|_2 \leq K_4$, we get that $\boldsymbol{a}^{(L)}_{i_L}$ is bounded along the whole SGD trajectory. Thus, we are able to conclude that $\left\|\boldsymbol{y}_j - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_j, \boldsymbol{\theta}(j)\right)\right\|_2 \leq K_5$, for some constant K_5.

We bound the absolute value of the increment $\|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j + 1) - \boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)\|_2$ as

$$\begin{aligned}\|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j + 1) - \boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)\|_2 &\leq 2\,\alpha\,\xi(j\alpha)\,N^2\,\left\|\boldsymbol{y}_j - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_j, \boldsymbol{\theta}(j)\right)\right\|_2 \\ &\quad\cdot \left\|\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op} \\ &\leq \alpha\,N^2\,C_1 \left\|\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op},\end{aligned} \tag{A.38}$$

where we use that ξ is bounded by K_1 and $\left\|\boldsymbol{y}_j - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_j, \boldsymbol{\theta}(j)\right)\right\|_2 \leq K_5$. Consequently,

$$\max_{i_\ell,i_{\ell+1}\in[N]} \|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j+1)-\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)\|_2 \leq \alpha\,N^2\,C_1 \max_{i_\ell,i_{\ell+1}\in[N]} \left\|\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op}. \tag{A.39}$$

Let us now focus on the operator norm of the Jacobian. First, we write

$$\begin{aligned}\left\|\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}}\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)\right\|_{op} &= \left\|\mathrm{D}_{\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}}\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)\cdot\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}}\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)\right\|_{op}\\ &\le \left\|\mathrm{D}_{\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}}\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)\right\|_{op}\cdot\left\|\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}}\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)\right\|_{op},\end{aligned}\tag{A.40}$$

where the inequality uses the fact that the operator norm is sub-multiplicative. Note that

$$\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}}\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)=\mathrm{diag}\left(\frac{1}{N}\sigma^{(\ell)}\left(\boldsymbol{z}^{(\ell)}_{i_\ell}\left(\boldsymbol{x},\boldsymbol{\theta}\right),\boldsymbol{w}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)\right)\right),\tag{A.41}$$

where we denote by $\mathrm{diag}(\boldsymbol{v})$ the diagonal matrix containing $\boldsymbol{v}$ on the diagonal. As $\sigma^{(\ell)}$ is bounded by assumption **(B3)**, we have that

$$\left\|\mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}}\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)\right\|_{op}\le\frac{C_2}{N}.\tag{A.42}$$

Furthermore, the Jacobian $\mathrm{D}_{\boldsymbol{z}^{(\ell+1)}_{i_{\ell+1}}}\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)$ is given by

$$\begin{aligned}\mathrm{D}_{\boldsymbol{z}^{(L)}_{i_L}}\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)&=\frac{1}{N}\boldsymbol{M}^{(L)}_{i_L}(\boldsymbol{x},\boldsymbol{\theta}(j)), \qquad i_L\in[N],\\ \mathrm{D}_{\boldsymbol{z}^{(\ell)}_{i_\ell}}\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x},\boldsymbol{\theta}(j)\right)&=\frac{1}{N^{L-\ell+1}}\sum_{\boldsymbol{p}^L_{\ell+1}\in[N]^{L-\ell}}\boldsymbol{M}^{(\ell)}_{i_\ell,\boldsymbol{p}^L_{\ell+1}}(\boldsymbol{x},\boldsymbol{\theta}(j)), \qquad \ell\in[L-1], i_\ell\in[N],\end{aligned}\tag{A.43}$$

where $\boldsymbol{p}^L_{\ell+1}$ denotes the multi-index $(p_{\ell+1},\dots,p_L)$, $[N]^{L-\ell}$ denotes the $(L-\ell)$-fold Cartesian product of $[N]$ and the matrices $\boldsymbol{M}^{(\ell)}_{\boldsymbol{p}^L_\ell}(\boldsymbol{x},\boldsymbol{\theta}(j))$ are defined recursively as

$$\begin{aligned}\boldsymbol{M}^{(L)}_{p_L}(\boldsymbol{x},\boldsymbol{\theta}(j))&=\mathrm{D}_{\boldsymbol{z}^{(L)}_{p_L}}\left(\boldsymbol{a}^{(L)}_{p_L}(j)\odot\sigma^{(L)}\left(\boldsymbol{z}^{(L)}_{p_L}(\boldsymbol{x},\boldsymbol{\theta}(j)),\boldsymbol{w}^{(L)}_{p_L}(j)\right)\right)\\ &=\mathrm{diag}(\boldsymbol{a}^{(L)}_{p_L}(j))\cdot\mathrm{D}_{\boldsymbol{z}^{(L)}_{p_L}}\sigma^{(L)}\left(\boldsymbol{z}^{(L)}_{p_L}(\boldsymbol{x},\boldsymbol{\theta}(j)),\boldsymbol{w}^{(L)}_{p_L}(j)\right),\\ \boldsymbol{M}^{(\ell)}_{\boldsymbol{p}^L_\ell}(\boldsymbol{x},\boldsymbol{\theta}(j))&=\boldsymbol{M}^{(\ell+1)}_{\boldsymbol{p}^L_{\ell+1}}(\boldsymbol{x},\boldsymbol{\theta}(j))\cdot\mathrm{D}_{\boldsymbol{z}^{(\ell)}_{p_\ell}}\left(\boldsymbol{a}^{(\ell)}_{p_\ell,p_{\ell+1}}(j)\odot\sigma^{(\ell)}\left(\boldsymbol{z}^{(\ell)}_{p_\ell}(\boldsymbol{x},\boldsymbol{\theta}(j)),\boldsymbol{w}^{(\ell)}_{p_\ell,p_{\ell+1}}(j)\right)\right)\\ &=\boldsymbol{M}^{(\ell+1)}_{\boldsymbol{p}^L_{\ell+1}}(\boldsymbol{x},\boldsymbol{\theta}(j))\cdot\mathrm{diag}(\boldsymbol{a}^{(\ell)}_{p_\ell,p_{\ell+1}}(j))\cdot\mathrm{D}_{\boldsymbol{z}^{(\ell)}_{p_\ell}}\sigma^{(\ell)}\left(\boldsymbol{z}^{(\ell)}_{p_\ell}(\boldsymbol{x},\boldsymbol{\theta}(j)),\boldsymbol{w}^{(\ell)}_{p_\ell,p_{\ell+1}}(j)\right).\end{aligned}\tag{A.44}$$

Note that $\boldsymbol{a}^{(L)}_{p_L}(j)=\boldsymbol{a}^{(L)}_{p_L}(0)$ and recall that $\|\boldsymbol{a}^{(L)}_{p_L}(0)\|_2$ is bounded by assumption **(B4)**. Furthermore, $\sigma^{(\ell)}$ has bounded Fréchet derivative by assumption **(B3)**. Thus, we deduce that

$$\left\|\boldsymbol{M}^{(L)}_{p_L}(\boldsymbol{x},\boldsymbol{\theta}(j))\right\|_{op}\le C_3,\tag{A.45}$$

and

$$\begin{aligned}\left\|\boldsymbol{M}^{(\ell)}_{\boldsymbol{p}^L_\ell}(\boldsymbol{x},\boldsymbol{\theta}(j))\right\|_{op}&\le C_4(L)\prod_{m=\ell}^{L-1}\|\boldsymbol{a}^{(m)}_{p_m,p_{m+1}}(j)\|_2\\ &\le C_4(L)\prod_{m=\ell}^{L-1}\max_{i_m,i_{m+1}\in[N]}\|\boldsymbol{a}^{(m)}_{i_m,i_{m+1}}(j)\|_2.\end{aligned}\tag{A.46}$$

Consequently, we have that

$$\begin{aligned}
&\left\| \mathrm{D}_{\boldsymbol{z}^{(L)}_{i_L}} \widehat{\boldsymbol{y}}_N \left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op} \le \frac{C_3}{N}, \\
&\left\| \mathrm{D}_{\boldsymbol{z}^{(\ell)}_{i_\ell}} \widehat{\boldsymbol{y}}_N \left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op} \le \frac{C_4(L)}{N} \prod_{m=\ell}^{L-1} \max_{i_m, i_{m+1}\in[N]} \|\boldsymbol{a}^{(m)}_{i_m,i_{m+1}}(j)\|_2.
\end{aligned} \tag{A.47}$$

By combining (A.40), (A.42) and (A.47), we obtain that

$$\begin{aligned}
&\left\| \mathrm{D}_{\boldsymbol{a}^{(L-1)}_{i_{L-1},i_L}} \widehat{\boldsymbol{y}}_N \left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op} \le \frac{C_5}{N^2}, \\
&\left\| \mathrm{D}_{\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N \left(\boldsymbol{x}, \boldsymbol{\theta}(j)\right)\right\|_{op} \le \frac{C_6(L)}{N^2} \prod_{m=\ell+1}^{L-1} \max_{i_m, i_{m+1}\in[N]} \|\boldsymbol{a}^{(m)}_{i_m,i_{m+1}}(j)\|_2, \qquad \ell \in [L-2].
\end{aligned} \tag{A.48}$$

By using also (A.39), we have that

$$\begin{aligned}
\max_{i_{L-1},i_L\in[N]} \|\boldsymbol{a}^{(L-1)}_{i_{L-1},i_L}(j+1) - \boldsymbol{a}^{(L-1)}_{i_{L-1},i_L}(j)\|_2 &\le \alpha\, C_7, \\
\max_{i_\ell,i_{\ell+1}\in[N]} \|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j+1) - \boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)\|_2 &\le \alpha\, C_8(L) \prod_{m=\ell+1}^{L-1} \max_{i_m, i_{m+1}\in[N]} \|\boldsymbol{a}^{(m)}_{i_m,i_{m+1}}(j)\|_2,
\end{aligned} \tag{A.49}$$

where $\ell \in [L-2]$.

By triangle inequality, we also obtain that, for $\ell \in [L-1]$ and $i_\ell, i_{\ell+1} \in [N]$,

$$\|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(s)\|_2 \le \sum_{j=0}^{s-1} \|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j+1) - \boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(j)\|_2 + \|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(0)\|_2. \tag{A.50}$$

As $\|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(0)\|_2$ and $\|\boldsymbol{a}^{(L)}_{i_L}(0)\|_2$ are bounded, by combining (A.49) and (A.50), we have that

$$\begin{aligned}
&\max_{s\in[k]} \max_{i_{L-1},i_L\in[N]} \|\boldsymbol{a}^{(L-1)}_{i_{L-1},i_L}(s)\|_2 \le C + C_7\, T, \\
&\max_{s\in[k]} \max_{i_\ell,i_{\ell+1}\in[N]} \|\boldsymbol{a}^{(\ell)}_{i_\ell,i_{\ell+1}}(s)\|_2 \le C + C_8(L)\, T \prod_{m=\ell+1}^{L-1} \max_{s\in[k]} \max_{i_m, i_{m+1}\in[N]} \|\boldsymbol{a}^{(m)}_{i_m,i_{m+1}}(s)\|_2,
\end{aligned} \tag{A.51}$$

where $\ell \in [L-2]$ and we have used that $T = k\alpha$. By doing a step of induction on $\ell \in \{L-2, L-3, \dots, 1\}$, the proof is complete. □

We are now ready to provide the proof of Theorem 2, part (A).

Proof of Theorem 2, part (A). For $\ell \in [L]$, we construct $\tilde{\sigma}^{(\ell)} : \mathbb{R}^{d_\ell} \times \mathbb{R}^{D_\ell + d_{\ell+1}} \to \mathbb{R}^{d_{\ell+1}}$ that satisfies the following two properties:

(i) $\tilde{\sigma}^{(\ell)}(\boldsymbol{z}, (\boldsymbol{w}, \boldsymbol{a}))$ coincides with $\boldsymbol{a} \odot \sigma^{(\ell)}(\boldsymbol{z}, \boldsymbol{w})$ for all $(\boldsymbol{z}, \boldsymbol{w}) \in \mathbb{R}^{d_\ell} \times \mathbb{R}^{D_\ell}$ and for $\|\boldsymbol{a}\|_2 \le K(T, L)$, where $K(T, L)$ is the bound of Lemma A.3.1;

(ii) $\tilde{\sigma}^{(\ell)}$ is bounded, with Fréchet derivatives bounded and Lipschitz.

Similarly, we construct $\tilde{\sigma}^{(L+1)} : \mathbb{R}^{d_{L+1}} \to \mathbb{R}^{d_{L+1}}$ that satisfies the following two properties:

(i) $\tilde{\sigma}^{(L+1)}(\boldsymbol{z}) = \boldsymbol{z}$ for $\|\boldsymbol{z}\|_2 \le K_3\,K_4$, where K_3 is the bound on $\sigma^{(L)}$ and K_4 is the bound on $\|\boldsymbol{a}_{i_L}^{(L)}(0)\|_2$ (see assumptions **(B3)**-**(B4)**);

(ii) $\tilde{\sigma}^{(L+1)}$ is bounded, with Fréchet derivatives bounded and Lipschitz.

Define

$$\begin{aligned}
(\boldsymbol{z}_{i_1}^{(1)})'(\boldsymbol{x},\boldsymbol{\theta}) &= \sigma^{(0)}\left(\boldsymbol{x},\boldsymbol{\theta}_{i_1}^{(0)}\right), \qquad i_1 \in [N],\\
(\boldsymbol{z}_{i_{\ell+1}}^{(\ell+1)})'(\boldsymbol{x},\boldsymbol{\theta}) &= \frac{1}{N}\sum_{i_\ell=1}^{N}\tilde{\sigma}^{(\ell)}\left((\boldsymbol{z}_{i_\ell}^{(\ell)})'(\boldsymbol{x},\boldsymbol{\theta}),\boldsymbol{\theta}_{i_\ell,i_{\ell+1}}^{(\ell)}\right), \qquad \ell\in[L-1],\, i_{\ell+1}\in[N],\\
\widehat{\boldsymbol{y}}'_N(\boldsymbol{x},\boldsymbol{\theta}) &= \tilde{\sigma}^{(L+1)}\left(\frac{1}{N}\sum_{i_L=1}^{N}\tilde{\sigma}^{(L)}\left((\boldsymbol{z}_{i_L}^{(L)})'(\boldsymbol{x},\boldsymbol{\theta}),\boldsymbol{\theta}_{i_L}^{(L)}\right)\right),
\end{aligned} \tag{A.52}$$

and

$$L'_N(\boldsymbol{\theta}) = \mathbb{E}\left\{\left\|\boldsymbol{y} - \widehat{\boldsymbol{y}}'_N(\boldsymbol{x},\boldsymbol{\theta})\right\|_2^2\right\}. \tag{A.53}$$

Let $\boldsymbol{\theta}'(k)$ be obtained by running k steps of the SGD algorithm (3.12) with $\widehat{\boldsymbol{y}}_N(\boldsymbol{x},\boldsymbol{\theta})$ replaced by $\widehat{\boldsymbol{y}}'_N(\boldsymbol{x},\boldsymbol{\theta})$. Recall that $\boldsymbol{a}_{i_\ell,i_{\ell+1}}^{(\ell)}(s)$ is bounded by Lemma A.3.1, $\boldsymbol{a}_{i_L}^{(L)}(s)$ is bounded by assumption **(B4)** and $\sigma^{(\ell)}$ is bounded by assumption **(B3)**. Thus, we have that $\boldsymbol{\theta}'(k) = \boldsymbol{\theta}(k)$ and $L'_N(\boldsymbol{\theta}'(k)) = L_N(\boldsymbol{\theta}(k))$. To simplify notation, in the rest of the proof we will drop the symbol $'$ from $\boldsymbol{\theta}$ and L_N. By definition of dropout stability, the proof is completed by showing that, with probability at least $1-e^{-z^2}$,

$$|L_N(\boldsymbol{\theta}(k)) - L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}(k))| \le K(T,L)\left(\frac{\sqrt{d}+z}{\sqrt{N}} + \sqrt{\alpha}\big(\sqrt{d}+z\big)\right). \tag{A.54}$$

By construction, the activation functions $\{\tilde{\sigma}^{(\ell)}\}_{\ell\in[L+1]}$ are bounded, with Fréchet derivatives that are bounded and Lipschitz. Thus, the technical assumptions of [AOY19] are fulfilled. Let $\rho^\star_{[0,T]}$ denote the unique solution to the McKean-Vlasov DNN problem with initial condition ρ_0 and activation functions $\sigma^{(0)}$ and $\tilde{\sigma}^{(\ell)}$, with $\ell\in\{0,\dots,L+1\}$. Furthermore, let $\bar{\boldsymbol{\theta}}(t)$, with $t\in[0,T]$, be the associated ideal particles. Furthermore, let $\bar{\boldsymbol{\theta}}_{\rm S}(t)$ be obtained from $\bar{\boldsymbol{\theta}}(t)$ in the same way in which $\boldsymbol{\theta}_{\rm S}(k)$ is obtained from $\boldsymbol{\theta}(k)$. By triangle inequality, we have that

$$\begin{aligned}
|L_N(\boldsymbol{\theta}(k)) - L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}(k))| &\le |L_N(\boldsymbol{\theta}(k)) - \bar{L}(\rho^\star_T)| + |L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}(k)) - \bar{L}(\rho^\star_T)|\\
&\le |L_N(\boldsymbol{\theta}(k)) - L_N(\bar{\boldsymbol{\theta}}(T))| + |L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\rm S}(k)) - L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}(T))|\\
&\quad + |L_N(\bar{\boldsymbol{\theta}}(T)) - \bar{L}(\rho^\star_T)| + |L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\rm S}(T)) - \bar{L}(\rho^\star_T)|,
\end{aligned} \tag{A.55}$$

where $\rho^\star_T$ denotes the marginal of $\rho^\star_{[0,T]}$ at time T and $\bar{L}$ is defined in (A.35).

Given a vector of parameters $\boldsymbol{\theta}$ containing N_ℓ neurons in layer ℓ ($\ell\in[L]$), we define the norm

$$\|\boldsymbol{\theta}\|_\infty = \max\left(\sup_{i_1\in[N_1]}\left\|\boldsymbol{\theta}_{i_1}^{(0)}\right\|_2, \sup_{\ell\in[L-1],i_\ell\in[N_\ell],i_{\ell+1}\in[N_{\ell+1}]}\left\|\boldsymbol{\theta}_{i_\ell,i_{\ell+1}}^{(\ell)}\right\|_2, \sup_{i_L\in[N_L]}\left\|\boldsymbol{\theta}_{i_L}^{(L)}\right\|_2\right). \tag{A.56}$$

As a preliminary result, we provide a bound on $\|\boldsymbol{\theta}(k) - \bar{\boldsymbol{\theta}}(T)\|_\infty$.

Consider the continuous time gradient descent process $\tilde{\boldsymbol{\theta}}(t)$, defined as

$$\begin{aligned}\tilde{\boldsymbol{\theta}}^{(0)}_{i_1}(t) &= \tilde{\boldsymbol{\theta}}^{(0)}_{i_1}(0),\\ \tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}(t) &= \tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}(0) + 2\int_0^t \alpha\xi(s)N^2\mathbb{E}\left\{\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}(s)\right)\right)^{\mathsf{T}} \mathrm{D}_{\tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}(s)\right)\right\}\mathrm{d}s,\\ \tilde{\boldsymbol{\theta}}^{(L)}_{i_L}(t) &= \tilde{\boldsymbol{\theta}}^{(L)}_{i_L}(0),\end{aligned} \tag{A.57}$$

with the initialization $\tilde{\boldsymbol{\theta}}^{(0)}_{i_1}(0) = \boldsymbol{\theta}^{(0)}_{i_1}(0)$, $\tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}(0) = \boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}(0)$ and $\tilde{\boldsymbol{\theta}}^{(L)}_{i_L}(0) = \boldsymbol{\theta}^{(L)}_{i_L}(0)$. By triangle inequality, we have that

$$\|\boldsymbol{\theta}(k) - \bar{\boldsymbol{\theta}}(T)\|_\infty \le \|\boldsymbol{\theta}(k) - \tilde{\boldsymbol{\theta}}(T)\|_\infty + \|\tilde{\boldsymbol{\theta}}(T) - \bar{\boldsymbol{\theta}}(T)\|_\infty. \tag{A.58}$$

In order to bound the first term in the RHS of (A.58), we follow a strategy similar to that of Proposition 10.1 in [AOY19]. From formula (10.8) of [AOY19], we have that

$$\begin{aligned}\left\|\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}(m) - \tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}(m\alpha)\right\|_2 &\le \alpha\left\|\mathrm{Mrt}^{(\ell)}_{i_\ell,i_{\ell+1}}(m)\right\|_2 +\\ &\sum_{r=1}^m \int_{(r-1)\alpha}^{r\alpha} \mathbb{E}\Bigg\{\bigg\|\alpha\xi((r-1)\alpha)\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(r-1)\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(r-1)\right)\\ &\qquad - \alpha\xi(s)\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}(s)\right)\right)^{\mathsf{T}} \mathrm{D}_{\tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}(s)\right)\bigg\|_2\Bigg\}\mathrm{d}s,\end{aligned} \tag{A.59}$$

where

$$\begin{aligned}\mathrm{Mrt}^{(\ell)}_{i_\ell,i_{\ell+1}}(m) = \sum_{r=1}^m \alpha\xi((r-1)\alpha)\Bigg(&\left(\boldsymbol{y}_{r-1} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_{r-1}, \boldsymbol{\theta}(r-1)\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}_{r-1}, \boldsymbol{\theta}(r-1)\right)\\ &- \mathbb{E}\Big\{\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(r-1)\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(r-1)\right)\Big\}\Bigg)\end{aligned} \tag{A.60}$$

is a martingale with respect to the filtration $\{\mathcal{F}_m,\ m \in \mathbb{N}\}$ with

$$\mathcal{F}_m = \sigma\Big(\boldsymbol{\theta}(0), (\boldsymbol{x}_0, \boldsymbol{y}_0), \dots, (\boldsymbol{x}_{m-1}, \boldsymbol{y}_{m-1})\Big).$$

By taking the sup on both sides, we have that

$$\begin{aligned}\sup_{\ell\in[L-1], i_\ell, i_{\ell+1}\in[N]} \left\|\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}(m) - \tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}(T)\right\|_2 &\le \overbrace{\alpha \sup_{\ell\in[L-1], i_\ell, i_{\ell+1}\in[N]} \left\|\mathrm{Mrt}^{(\ell)}_{i_\ell,i_{\ell+1}}(m)\right\|_2}^{\textit{(I)}} +\\ &\overbrace{\begin{aligned}\sum_{r=1}^m \int_{(r-1)\alpha}^{r\alpha} \mathbb{E}\Bigg\{\sup_{\ell\in[L-1], i_\ell, i_{\ell+1}\in[N]} \bigg\|&\alpha\xi((r-1)\alpha)\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(r-1)\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}(r-1)\right)\\ &- \alpha\xi(s)\left(\boldsymbol{y} - \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}(s)\right)\right)^{\mathsf{T}} \mathrm{D}_{\tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell,i_{\ell+1}}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}(s)\right)\bigg\|_2\Bigg\}\mathrm{d}s.\end{aligned}}^{\textit{(II)}}\end{aligned} \tag{A.61}$$

Given two parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, by following the argument of Lemma B.17 of [AOY19], we have that

$$\left\|\left(\boldsymbol{y}-\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}_1\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}_{i_\ell, i_{\ell+1}}^{(\ell)}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}_1\right)-\left(\boldsymbol{y}-\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}_2\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}_{i_\ell, i_{\ell+1}}^{(\ell)}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}_2\right)\right\|_2 \\ \leq C_1\|\boldsymbol{\theta}_1-\boldsymbol{\theta}_2\|_\infty. \tag{A.62}$$

In what follows, the C_i are constants that depend on L, T, and on the constants K_i of the assumptions.

Consequently, we can bound the second term in the RHS of (A.61) as

$$\textit{(II)} \leq C_2 \sum_{r=1}^{m} \int_{(r-1)\varepsilon}^{r\varepsilon}\Big(|(r-1)\varepsilon-s|+\|\boldsymbol{\theta}(r-1)-\tilde{\boldsymbol{\theta}}(s)\|_\infty\Big)\mathrm{d}s, \tag{A.63}$$

where we have used that the quantity

$$\left(\boldsymbol{y}-\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}\right)\right)^{\mathsf{T}} \mathrm{D}_{\boldsymbol{\theta}_{i_\ell, i_{\ell+1}}^{(\ell)}} \widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \boldsymbol{\theta}\right) \tag{A.64}$$

is bounded for all $\boldsymbol{\theta}$. By using also that the process $t \to \tilde{\boldsymbol{\theta}}(t)$ is Lipschitz in time, we obtain the bound

$$\textit{(II)} \leq C_3\, \alpha\, T + C_3\, \alpha \sum_{r=0}^{m-1}\left\|\boldsymbol{\theta}(r)-\tilde{\boldsymbol{\theta}}(r)\right\|_\infty. \tag{A.65}$$

By combining (A.65) with (A.61) and by applying a discrete Gronwall inequality, we have that

$$\left\|\boldsymbol{\theta}(k)-\tilde{\boldsymbol{\theta}}(T)\right\|_\infty \leq \alpha\, e^{C_3 T}\left(\sup_{m\in[k]}\|\mathrm{Mrt}(m)\|_\infty + C_3\, T\right), \tag{A.66}$$

where we have defined

$$\|\mathrm{Mrt}(m)\|_\infty = \sup_{\ell\in[L-1], i_\ell, i_{\ell+1}\in[N]}\left\|\mathrm{Mrt}_{i_\ell, i_{\ell+1}}^{(\ell)}(m)\right\|_2. \tag{A.67}$$

Note that $e^{\zeta\|\mathrm{Mrt}(m)\|_\infty}$ is a submartingale. By using a Cramér-Chernoff argument, we have that

$$\mathbb{P}\left(\sup_{m\in[k]}\|\mathrm{Mrt}_N(m)\|_\infty > u\right) \leq \inf_{\zeta\in\mathbb{R}^+} e^{-\zeta\cdot u}\,\mathbb{E}\left\{e^{\zeta\|\mathrm{Mrt}(\tau)\|_\infty}\right\} \leq \inf_{\zeta\in\mathbb{R}^+} e^{-\zeta\cdot u}\,\mathbb{E}\left\{e^{\zeta\|\mathrm{Mrt}(k)\|_\infty}\right\}, \tag{A.68}$$

where $\tau = \inf\{m \leq k, \|\mathrm{Mrt}_N(m)\|_\infty > u\} \wedge k$ is a stopping time, and in the second inequality we have applied the optional stopping theorem to the submartingale $e^{\zeta\|\mathrm{Mrt}(m)\|_\infty}$. Furthermore, for any $\zeta > 0$, we have that

$$\mathbb{E}\left\{e^{\zeta\|\mathrm{Mrt}(k)\|_\infty}\right\} \leq \sum_{\ell=1}^{L}\sum_{i_\ell, i_{\ell+1}=1}^{N}\mathbb{E}\left\{e^{\zeta\left\|\mathrm{Mrt}_{i_\ell, i_{\ell+1}}^{(\ell)}(k)\right\|_2}\right\}. \tag{A.69}$$

Note that the martingale $\mathrm{Mrt}_{i_\ell, i_{\ell+1}}^{(\ell)}(k)$ has bounded increments. Thus, by using a modification of Hoeffding's Lemma and an ε-net argument (cf. Lemma A.3 of [AOY19]), we obtain that

$$\mathbb{E}\left\{e^{\zeta\left\|\mathrm{Mrt}_{i_\ell i_{\ell+1}}^{(\ell)}(k)\right\|_2}\right\} \leq 5^d \cdot e^{C_4 k \zeta^2}, \tag{A.70}$$

with $d = \max_{i \in [L-1]} d_i$. By combining (A.68), (A.69) and (A.70), we deduce that

$$\mathbb{P}\left(\sup_{m \in [k]} \|\mathrm{Mrt}_N(m)\|_\infty > u\right) \leq L N^2 5^d \inf_{\zeta \in \mathbb{R}^+} e^{-\zeta\, u + C_4 k \zeta^2}. \tag{A.71}$$

By optimizing over ζ, we have that, with probability at least $1 - e^{-z^2}$,

$$\sup_{m \in [k]} \|\mathrm{Mrt}_N(m)\|_\infty \leq C_5 \sqrt{\frac{1}{\alpha}} \left(\sqrt{d + \log N} + z\right). \tag{A.72}$$

Finally, by combining (A.72) with (A.66), we conclude that, with probability at least $1 - e^{-z^2}$,

$$\left\|\boldsymbol{\theta}(k) - \tilde{\boldsymbol{\theta}}(T)\right\|_\infty \leq C_6 \sqrt{\alpha}(\sqrt{d + \log N} + z). \tag{A.73}$$

Let us bound the second term in the RHS of (A.58). By following the strategy of Lemma 12.2 in [AOY19], we have that, with probability at least $1 - e^{-u^2}$,

$$\left\|\tilde{\boldsymbol{\theta}}^{(\ell)}_{i_\ell, i_{\ell+1}}(t) - \overline{\boldsymbol{\theta}}^{(\ell)}_{i_\ell, i_{\ell+1}}(t)\right\|_2 \leq C_7 \int_0^t \left\|\tilde{\boldsymbol{\theta}}(s) - \overline{\boldsymbol{\theta}}(s)\right\|_\infty \mathrm{d}s + C_7 \frac{u + \sqrt{d}}{\sqrt{N}}. \tag{A.74}$$

By doing a union bound over $i_\ell, i_{\ell+1} \in [N]$ and $\ell \in [L-1]$, we deduce that, with probability at least $1 - e^{-z^2}$,

$$\left\|\tilde{\boldsymbol{\theta}}(t) - \overline{\boldsymbol{\theta}}(t)\right\|_\infty \leq C_7 \int_0^t \left\|\tilde{\boldsymbol{\theta}}(s) - \overline{\boldsymbol{\theta}}(s)\right\|_\infty \mathrm{d}s + C_8 \frac{z + \sqrt{d + \log N}}{\sqrt{N}}. \tag{A.75}$$

By Gronwall lemma, we conclude that, with probability at least $1 - e^{-z^2}$,

$$\left\|\tilde{\boldsymbol{\theta}}(T) - \overline{\boldsymbol{\theta}}(T)\right\|_\infty \leq C_8 e^{C_7 T} \frac{z + \sqrt{d + \log N}}{\sqrt{N}}. \tag{A.76}$$

By combining (A.73) and (A.76), we have that, with probability at least $1 - e^{-z^2}$,

$$\|\boldsymbol{\theta}(k) - \bar{\boldsymbol{\theta}}(T)\|_\infty \leq C_9 \left(\frac{z + \sqrt{d + \log N}}{\sqrt{N}} + \sqrt{\alpha}(\sqrt{d + \log N} + z)\right). \tag{A.77}$$

At this point, we are ready to bound the various terms in the RHS of (A.55). In order to bound the first term, note that L_N is Lipschitz with $\|\cdot\|_\infty$. Thus, we obtain that, with probability at least $1 - e^{-z^2}$,

$$|L_N(\boldsymbol{\theta}(k)) - L_N(\bar{\boldsymbol{\theta}}(T))| \leq C_{10} \left(\frac{z + \sqrt{d + \log N}}{\sqrt{N}} + \sqrt{\alpha}(\sqrt{d + \log N} + z)\right). \tag{A.78}$$

In order to bound the second term in the RHS of (A.55), note that

$$\|\boldsymbol{\theta}_{\mathrm{S}}(k) - \bar{\boldsymbol{\theta}}_{\mathrm{S}}(T)\|_\infty \leq \|\boldsymbol{\theta}(k) - \bar{\boldsymbol{\theta}}(T)\|_\infty. \tag{A.79}$$

As $L_{|\mathcal{A}|}$ is Lipschitz with $\|\cdot\|_\infty$, by combining (A.77) and (A.79), we obtain the bound

$$|L_{|\mathcal{A}|}(\boldsymbol{\theta}_{\mathrm{S}}(k)) - L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\mathrm{S}}(T))| \leq C_{11} \left(\frac{z + \sqrt{d + \log N}}{\sqrt{N}} + \sqrt{\alpha}(\sqrt{d + \log N} + z)\right), \tag{A.80}$$

with probability at least $1 - e^{-z^2}$.

Finally, let us consider the remaining two terms in the RHS of (A.55). Fix $\boldsymbol{x} \in \mathbb{R}^{d_0}$. Then, by Lemma 11.4 of [AOY19], we have that, for $\zeta > 0$,

$$\log \mathbb{E}\left\{ e^{\zeta \|\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \bar{\boldsymbol{\theta}}(T)\right) - \bar{\boldsymbol{y}}(\boldsymbol{x}, \rho_T^\star)\|_2} \right\} \le C_{12}\left(d + \frac{\zeta^2}{N}\right). \tag{A.81}$$

By using similar arguments, we also have that, for $\zeta > 0$,

$$\log \mathbb{E}\left\{ e^{\zeta \|\widehat{\boldsymbol{y}}_{|\mathcal{A}|}\left(\boldsymbol{x}, \bar{\boldsymbol{\theta}}_{\mathrm{S}}(T)\right) - \bar{\boldsymbol{y}}(\boldsymbol{x}, \rho_T^\star)\|_2} \right\} \le C_{13}\left(d + \frac{\zeta^2}{A_{\min}}\right). \tag{A.82}$$

Thus, by applying Markov inequality and optimizing over ζ, we deduce that

$$\begin{aligned} \|\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \bar{\boldsymbol{\theta}}(T)\right) - \bar{\boldsymbol{y}}(\boldsymbol{x}, \rho_T^\star)\|_2 &\le C_{14}\frac{\sqrt{d} + z}{\sqrt{N}}, \\ \|\widehat{\boldsymbol{y}}_{|\mathcal{A}|}\left(\boldsymbol{x}, \bar{\boldsymbol{\theta}}_{\mathrm{S}}(T)\right) - \bar{\boldsymbol{y}}(\boldsymbol{x}, \rho_T^\star)\|_2 &\le C_{14}\frac{\sqrt{d} + z}{\sqrt{A_{\min}}}, \end{aligned} \tag{A.83}$$

with probability at least $1 - e^{-z^2}$. By using that $\boldsymbol{y}$, $\widehat{\boldsymbol{y}}_N\left(\boldsymbol{x}, \bar{\boldsymbol{\theta}}(T)\right)$, $\widehat{\boldsymbol{y}}_{|\mathcal{A}|}\left(\boldsymbol{x}, \bar{\boldsymbol{\theta}}_{\mathrm{S}}(T)\right)$ and $\bar{\boldsymbol{y}}(\boldsymbol{x}, \rho_T^\star)$ are bounded, we conclude that

$$\begin{aligned} |L_N(\bar{\boldsymbol{\theta}}(T)) - \bar{L}(\rho_T^\star)| &\le C_{15}\frac{\sqrt{d} + z}{\sqrt{N}}, \\ |L_{|\mathcal{A}|}(\bar{\boldsymbol{\theta}}_{\mathrm{S}}(T)) - \bar{L}(\rho_T^\star)| &\le C_{15}\frac{\sqrt{d} + z}{\sqrt{A_{\min}}}, \end{aligned} \tag{A.84}$$

with probability at least $1 - e^{-z^2}$. By combining (A.78), (A.80) and (A.84), the proof is complete. □

A.3.2 Part (B)

The proof of part (B) is obtained by combining part (A) with the following result, which extends Lemma A.1.1 to the multilayer case.

Lemma A.3.2 (Dropout stability implies connectivity – multilayer). *Consider a neural network with $L + 1 \ge 4$ layers, where each hidden layer contains N neurons, as in* (3.10). *For any $k \in [L]$, assume that $\boldsymbol{\theta}$ and $\bar{\boldsymbol{\theta}}$ are ε-dropout stable given $\mathcal{A}_i = [N/2]$ for $i \in \{k, \dots, L\}$. Then, $\boldsymbol{\theta}$ and $\bar{\boldsymbol{\theta}}$ are ε-connected.*

Given a vector of parameters $\boldsymbol{\theta}$, it is helpful to write it as

$$\begin{aligned} \boldsymbol{\theta}^{(L)} &= \left\{ \left[\boldsymbol{a}_{i_L}^{(L)}\right]_{i_L \in [N]}, \left[\boldsymbol{w}_{i_L}^{(L)}\right]_{i_L \in [N]} \right\}, \\ \boldsymbol{\theta}^{(\ell)} &= \left\{ \left[\boldsymbol{a}_{i_{\ell+1}, i_\ell}^{(\ell)}\right]_{i_{\ell+1}, i_\ell \in [N]}, \left[\boldsymbol{w}_{i_{\ell+1}, i_\ell}^{(\ell)}\right]_{i_{\ell+1}, i_\ell \in [N]} \right\}, \qquad \ell \in [L-1], \\ \boldsymbol{\theta}^{(0)} &= \left[\boldsymbol{\theta}_{i_0}^{(0)}\right]_{i_0 \in [N]}. \end{aligned} \tag{A.85}$$

In words, we stack the parameters $\boldsymbol{\theta}^{(\ell)}$ of layer ℓ into a matrix, and the (i,j)-th element of this matrix contains the parameter $\boldsymbol{\theta}_{j,i}^{(\ell)} = (\boldsymbol{a}_{j,i}^{(\ell)}, \boldsymbol{w}_{j,i}^{(\ell)})$ connecting the j-th neuron of layer ℓ with the i-th neuron of layer $\ell+1$. Furthermore, let us partition the parameters $\boldsymbol{\theta}$ as

$$\begin{aligned}
\boldsymbol{\theta}^{(L)} &= \left\{ \left[\ \boldsymbol{a}_{\mathrm{t}}^{(L)} \;\middle|\; \boldsymbol{a}_{\mathrm{b}}^{(L)}\ \right], \left[\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \;\middle|\; \boldsymbol{w}_{\mathrm{b}}^{(L)}\ \right] \right\}, \\
\boldsymbol{\theta}^{(\ell)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{a}_{\mathrm{t,t}}^{(\ell)} & \boldsymbol{a}_{\mathrm{t,b}}^{(\ell)} \\ \hline \boldsymbol{a}_{\mathrm{b,t}}^{(\ell)} & \boldsymbol{a}_{\mathrm{b,b}}^{(\ell)} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(\ell)} & \boldsymbol{w}_{\mathrm{t,b}}^{(\ell)} \\ \hline \boldsymbol{w}_{\mathrm{b,t}}^{(\ell)} & \boldsymbol{w}_{\mathrm{b,b}}^{(\ell)} \end{array}\right] \right\}, \qquad \ell \in [L-1], \\
\boldsymbol{\theta}^{(0)} &= \left[\begin{array}{c} \boldsymbol{\theta}_{\mathrm{t}}^{(0)} \\ \hline \boldsymbol{\theta}_{\mathrm{b}}^{(0)} \end{array}\right].
\end{aligned} \tag{A.86}$$

In words, $\boldsymbol{\theta}_{\mathrm{t,t}}^{(\ell)} = (\boldsymbol{a}_{\mathrm{t,t}}^{(\ell)}, \boldsymbol{w}_{\mathrm{t,t}}^{(\ell)})$ contains the parameters connecting the top half neurons of layer ℓ with the top half neurons of layer $\ell+1$; $\boldsymbol{\theta}_{\mathrm{t,b}}^{(\ell)} = (\boldsymbol{a}_{\mathrm{t,b}}^{(\ell)}, \boldsymbol{w}_{\mathrm{t,b}}^{(\ell)})$ contains the parameters connecting the bottom half neurons of layer ℓ with the top half neurons of layer $\ell+1$; $\boldsymbol{\theta}_{\mathrm{b,t}}^{(\ell)} = (\boldsymbol{a}_{\mathrm{b,t}}^{(\ell)}, \boldsymbol{w}_{\mathrm{b,t}}^{(\ell)})$ contains the parameters connecting the top half neurons of layer ℓ with the bottom half neurons of layer $\ell+1$; and $\boldsymbol{\theta}_{\mathrm{b,b}}^{(\ell)} = (\boldsymbol{a}_{\mathrm{b,b}}^{(\ell)}, \boldsymbol{w}_{\mathrm{b,b}}^{(\ell)})$ contains the parameters connecting the bottom half neurons of layer ℓ with the bottom half neurons of layer $\ell+1$. The partition for the first and the last layer is similarly defined.

At this point, we are ready to present the proof of Lemma A.3.2.

Proof of Lemma A.3.2. For the moment, assume that N is even. Let $\boldsymbol{\theta}_{\mathrm{S},k}$ be obtained from $\boldsymbol{\theta}$ by keeping only the top half neurons at layer $\ell \in \{k, \dots, L\}$. With an abuse of notation, we can partition the parameters $\boldsymbol{\theta}_{\mathrm{S},k}$ as

$$\begin{aligned}
\boldsymbol{\theta}_{\mathrm{S},k}^{(L)} &= \left\{ \left[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \;\middle|\; \boldsymbol{0}\ \right], \left[\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \;\middle|\; \boldsymbol{0}\ \right] \right\}, \\
\boldsymbol{\theta}_{\mathrm{S},k}^{(\ell)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(\ell)} & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{0} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(\ell)} & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{0} \end{array}\right] \right\}, \qquad \ell \in \{k, \dots, L-1\}, \\
\boldsymbol{\theta}_{\mathrm{S},k}^{(\ell)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{a}_{\mathrm{t,t}}^{(\ell)} & \boldsymbol{a}_{\mathrm{t,b}}^{(\ell)} \\ \hline \boldsymbol{a}_{\mathrm{b,t}}^{(\ell)} & \boldsymbol{a}_{\mathrm{b,b}}^{(\ell)} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(\ell)} & \boldsymbol{w}_{\mathrm{t,b}}^{(\ell)} \\ \hline \boldsymbol{w}_{\mathrm{b,t}}^{(\ell)} & \boldsymbol{w}_{\mathrm{b,b}}^{(\ell)} \end{array}\right] \right\}, \qquad \ell \in [k-1], \\
\boldsymbol{\theta}_{\mathrm{S},k}^{(0)} &= \left[\begin{array}{c} \boldsymbol{\theta}_{\mathrm{t}}^{(0)} \\ \hline \boldsymbol{\theta}_{\mathrm{b}}^{(0)} \end{array}\right],
\end{aligned} \tag{A.87}$$

and the corresponding loss is given by $L_N(\boldsymbol{\theta}_{\mathrm{S},k})$. We now prove by induction that $\boldsymbol{\theta}$ is connected to $\boldsymbol{\theta}_{\mathrm{S},k}$ via a piecewise linear path in parameter space, such that the loss along the path is upper bounded by $L_N(\boldsymbol{\theta}) + \varepsilon$.

Base step: from $\boldsymbol{\theta}$ to $\boldsymbol{\theta}_{\mathrm{S},L}$. As $\boldsymbol{\theta}$ is ε-dropout stable, we have that $L_N(\boldsymbol{\theta}_{\mathrm{S},L}) \le L_N(\boldsymbol{\theta}) + \varepsilon$. Note that if $\boldsymbol{a}_{\mathrm{t}}^{(L)} = \boldsymbol{b}0$, then the value of $\boldsymbol{w}_{\mathrm{t}}^{(L)}$ does not affect the loss. Hence, we can interpolate from $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)}\ \mid \boldsymbol{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)}\ \mid \boldsymbol{0}\ ]\}$ to $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)}\ \mid \boldsymbol{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)}\ \mid \boldsymbol{w}_{\mathrm{b}}^{(L)}\ ]\}$ with no change in loss. Furthermore, the loss is convex in $\boldsymbol{a}^{(L)}$. Thus, we can interpolate from $\{[\ \boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \boldsymbol{a}_{\mathrm{b}}^{(L)}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{w}_{\mathrm{b}}^{(L)}\ ]\}$ to $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \boldsymbol{b}0\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{w}_{\mathrm{b}}^{(L)}\ ]\}$ while keeping the loss upper bounded by $L_N(\boldsymbol{\theta}) + \varepsilon$.

Induction step: from $\boldsymbol{\theta}_{\mathrm{S},k}$ to $\boldsymbol{\theta}_{\mathrm{S},k-1}$. We construct the path by passing through the following intermediate points in parameter space:

$$
\begin{aligned}
\boldsymbol{\theta}_1^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right] \right\}, \\
\boldsymbol{\theta}_1^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right] \right\}, \qquad i \in \{k, \ldots, L-1\}, \\
\boldsymbol{\theta}_1^{(k-1)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{a}_{\mathrm{t,b}}^{(k-1)} \\ \hline \boldsymbol{a}_{\mathrm{b,t}}^{(k-1)} & \boldsymbol{a}_{\mathrm{b,b}}^{(k-1)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{w}_{\mathrm{t,b}}^{(k-1)} \\ \hline \boldsymbol{w}_{\mathrm{b,t}}^{(k-1)} & \boldsymbol{w}_{\mathrm{b,b}}^{(k-1)} \end{array} \right] \right\}.
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\theta}_2^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right] \right\}, \\
\boldsymbol{\theta}_2^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{w}_{\mathrm{t,t}}^{(i)} \end{array} \right] \right\}, \qquad i \in \{k, \ldots, L-1\}, \\
\boldsymbol{\theta}_2^{(k-1)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{a}_{\mathrm{t,b}}^{(k-1)} \\ \hline 2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{w}_{\mathrm{t,b}}^{(k-1)} \\ \hline \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right] \right\}.
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\theta}_3^{(L)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{b}0 & 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{b}0 & \boldsymbol{w}_{\mathrm{t}}^{(L)} \end{array} \right] \right\}, \\
\boldsymbol{\theta}_3^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{w}_{\mathrm{t,t}}^{(i)} \end{array} \right] \right\}, \qquad i \in \{k, \ldots, L-1\}, \\
\boldsymbol{\theta}_3^{(k-1)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{a}_{\mathrm{t,b}}^{(k-1)} \\ \hline 2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{w}_{\mathrm{t,b}}^{(k-1)} \\ \hline \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right] \right\}.
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\theta}_4^{(L)} &= \left\{ \left[\begin{array}{c|c} \boldsymbol{b}0 & 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{b}0 & \boldsymbol{w}_{\mathrm{t}}^{(L)} \end{array} \right] \right\}, \\
\boldsymbol{\theta}_4^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{w}_{\mathrm{t,t}}^{(i)} \end{array} \right] \right\}, \qquad i \in \{k, \ldots, L-1\}, \\
\boldsymbol{\theta}_4^{(k-1)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{b}0 \\ \hline 2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{b}0 \\ \hline \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right] \right\}.
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\theta}_5^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right] \right\}, \\
\boldsymbol{\theta}_5^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{w}_{\mathrm{t,t}}^{(i)} \end{array} \right] \right\}, \qquad i \in \{k, \ldots, L-1\}, \\
\boldsymbol{\theta}_5^{(k-1)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{b}0 \\ \hline 2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \boldsymbol{b}0 \\ \hline \boldsymbol{w}_{\mathrm{t,t}}^{(k-1)} & \mathbf{0} \end{array} \right] \right\}.
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\theta}_6^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array} \right] \right\}, \\
\boldsymbol{\theta}_6^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array} \right] \right\}, \qquad i \in \{k-1, \ldots, L-1\}.
\end{aligned}
$$

As we do not change the parameters in layer $\ell \in [k-2]$, we have omitted them in the definitions above.

From $\boldsymbol{\theta}_1$ to $\boldsymbol{\theta}_2$. The loss is not affected by the values in the bottom right quadrant of $\boldsymbol{\theta}_1^{(k-1)}$, since the bottom neurons of layer k are not active ($\boldsymbol{a}_{\mathrm{t,b}}^{(k)} = \boldsymbol{a}_{\mathrm{b,b}}^{(k)} = \boldsymbol{b}0$). Consequently, we can interpolate from $\boldsymbol{a}_{\mathrm{b,b}}^{(k-1)}$ to $\boldsymbol{b}0$ and from $\boldsymbol{w}_{\mathrm{b,b}}^{(k-1)}$ to $\boldsymbol{b}0$ with no change in loss. Similarly, the loss is not affected by the values in the bottom right quadrant of $\boldsymbol{\theta}_1^{(i)}$ for $i \in \{k, \dots L-1\}$, since the bottom neurons of layer $i+1$ are not active ($\boldsymbol{a}_{\mathrm{t,b}}^{(i+1)} = \boldsymbol{a}_{\mathrm{b,b}}^{(i+1)} = \boldsymbol{b}0$ and $\boldsymbol{a}_{\mathrm{b}}^{(L)} = \boldsymbol{b}0$). Consequently, for $i \in \{k, \dots L-1\}$, we can successively interpolate from $\boldsymbol{b}0$ to $2\boldsymbol{a}_{\mathrm{t,t}}^{(i)}$ and from $\boldsymbol{b}0$ to $2\boldsymbol{w}_{\mathrm{t,t}}^{(i)}$ with no change in loss.

From $\boldsymbol{\theta}_5$ to $\boldsymbol{\theta}_6$. We use the same reasoning as for $\boldsymbol{\theta}_1 \to \boldsymbol{\theta}_2$ and go in reverse layer order (i.e., from layer $L-1$ to layer $k-1$). The loss is not affected by the values in the bottom right quadrant of $\boldsymbol{\theta}_5^{(i)}$, since the bottom neurons of layer $i+1$ are not active. Consequently, we can interpolate from $2\boldsymbol{a}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ and from $\boldsymbol{w}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ with no change in loss. Similarly, the loss is not affected by the values in the bottom left quadrant of $\boldsymbol{\theta}_5^{(k-1)}$, since the bottom neurons of layer k are not active. Consequently, we can interpolate from $2\boldsymbol{a}_{\mathrm{t,t}}^{(k-1)}$ to $\boldsymbol{b}0$ and from $\boldsymbol{w}_{\mathrm{t,t}}^{(k-1)}$ to $\boldsymbol{b}0$ with no change in loss.

From $\boldsymbol{\theta}_4$ to $\boldsymbol{\theta}_5$. Note that the parameters of $\boldsymbol{\theta}_4$ and $\boldsymbol{\theta}_5$ are the same except for layer L. Furthermore, the structure of these parameters implies that the output of layer $L-1$ is obtained by stacking the output of two identical sub-networks. In formulas, let $\boldsymbol{z}^{(L-1)}$ be the output of layer $L-1$. Then, $\boldsymbol{z}^{(L-1)} = [\ \bar{\boldsymbol{z}} \mid \bar{\boldsymbol{z}}\ ]$ for some $\bar{\boldsymbol{z}}$. Consequently, we can interpolate between $\boldsymbol{\theta}_4$ and $\boldsymbol{\theta}_5$ with no change in loss.

From $\boldsymbol{\theta}_3$ to $\boldsymbol{\theta}_4$. By using the same reasoning as for $\boldsymbol{\theta}_5 \to \boldsymbol{\theta}_6$, we interpolate from $2\boldsymbol{a}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ and from $\boldsymbol{w}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ in the top left corner of $\boldsymbol{\theta}_3^{(i)}$ with no change in loss, for $i = L-1, \dots, k$. Then, we interpolate from $\boldsymbol{\theta}_3^{(k-1)}$ to $\boldsymbol{\theta}_4^{(k-1)}$ with no change in loss, since the top neurons of layer k are not active. Finally, we restore sequentially $2\boldsymbol{a}_{\mathrm{t,t}}^{(i)}$ and $\boldsymbol{w}_{\mathrm{t,t}}^{(i)}$ in the top left corner of the corresponding parameter matrices with no change in loss, by using the same reasoning as for $\boldsymbol{\theta}_1 \to \boldsymbol{\theta}_2$.

From $\boldsymbol{\theta}_2$ to $\boldsymbol{\theta}_3$. From the previous arguments, we have that $L_N(\boldsymbol{\theta}_2) = L_N(\boldsymbol{\theta}_1)$ and $L_N(\boldsymbol{\theta}_3) = L_N(\boldsymbol{\theta}_6)$. Furthermore, $\boldsymbol{\theta}$ is ε-dropout stable, which implies that $|L_N(\boldsymbol{\theta}_1) - L_N(\boldsymbol{\theta}_6)| \leq \varepsilon$. Consequently, we have that $|L_N(\boldsymbol{\theta}_2) - L_N(\boldsymbol{\theta}_3)| \leq \varepsilon$. Note that if $\boldsymbol{a}_{\mathrm{t}}^{(L)} = \boldsymbol{b}0$, then the value of $\boldsymbol{w}_{\mathrm{t}}^{(L)}$ does not affect the loss. Hence, we can interpolate from $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \mathbf{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \mathbf{0}\ ]\}$ to $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \mathbf{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{w}_{\mathrm{t}}^{(L)}\ ]\}$ with no change in loss. Similarly, we can interpolate from $\{[\ \boldsymbol{b}0 \mid 2\boldsymbol{a}_{\mathrm{t}}^{(L)}\ ], [\ \boldsymbol{b}0 \mid \boldsymbol{w}_{\mathrm{t}}^{(L)}\ ]\}$ to $\{[\ \boldsymbol{b}0 \mid 2\boldsymbol{a}_{\mathrm{t}}^{(L)}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{w}_{\mathrm{t}}^{(L)}\ ]\}$ with no change in loss. Furthermore, the loss is convex in $\boldsymbol{a}^{(L)}$. Thus, we can interpolate from $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \mathbf{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{w}_{\mathrm{t}}^{(L)}\ ]\}$ to $\{[\ \boldsymbol{b}0 \mid 2\boldsymbol{a}_{\mathrm{t}}^{(L)}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{w}_{\mathrm{t}}^{(L)}\ ]\}$ while keeping the loss upper bounded by $L_N(\boldsymbol{\theta}) + \varepsilon$.

As a result, we are able to connect $\boldsymbol{\theta}$ with $\boldsymbol{\theta}_{\mathrm{S},1}$ via a piecewise linear path, where the loss is upper bounded by $L_N(\boldsymbol{\theta}) + \varepsilon$. Similarly, let $\bar{\boldsymbol{\theta}}_{\mathrm{S},k}$ be obtained from $\bar{\boldsymbol{\theta}}$ by keeping only the top half neurons at layer $\ell \in \{k, \dots, L\}$. Then, we can connect $\bar{\boldsymbol{\theta}}$ with $\bar{\boldsymbol{\theta}}_{\mathrm{S},1}$ via a piecewise linear path, where the loss is upper bounded by $L_N(\bar{\boldsymbol{\theta}}) + \varepsilon$.

In order to complete the proof, it remains to connect $\boldsymbol{\theta}_{\mathrm{S},1}$ with $\bar{\boldsymbol{\theta}}_{\mathrm{S},1}$ via a piecewise linear path, where the loss is upper bounded by $\max(L_N(\boldsymbol{\theta}), L_N(\bar{\boldsymbol{\theta}})) + \varepsilon$. We construct the path by passing through the following intermediate points in parameter space:

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}_1^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array}\right] \right\}, \\
\tilde{\boldsymbol{\theta}}_1^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \right\}, \qquad i \in \{1, \dots, L-1\}, \\
\tilde{\boldsymbol{\theta}}_1^{(0)} &= \left[\begin{array}{c} \boldsymbol{\theta}_{\mathrm{t}}^{(0)} \\ \hline \boldsymbol{\theta}_{\mathrm{b}}^{(0)} \end{array}\right].
\end{aligned}$$

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}_2^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array}\right] \right\}, \\
\tilde{\boldsymbol{\theta}}_2^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} \end{array}\right] \right\}, \qquad i \in \{1, \dots, L-1\}, \\
\tilde{\boldsymbol{\theta}}_2^{(0)} &= \left[\begin{array}{c} \boldsymbol{\theta}_{\mathrm{t}}^{(0)} \\ \hline \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \end{array}\right].
\end{aligned}$$

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}_3^{(L)} &= \left\{ \left[\begin{array}{c|c} \mathbf{0} & 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{b}0 & \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)} \end{array}\right] \right\}, \\
\tilde{\boldsymbol{\theta}}_3^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\boldsymbol{a}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{w}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} \end{array}\right] \right\}, \qquad i \in \{1, \dots, L-1\}, \\
\tilde{\boldsymbol{\theta}}_3^{(0)} &= \left[\begin{array}{c} \boldsymbol{\theta}_{\mathrm{t}}^{(0)} \\ \hline \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \end{array}\right].
\end{aligned}$$

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}_4^{(L)} &= \left\{ \left[\begin{array}{c|c} \mathbf{0} & 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)} \end{array}\right], \left[\begin{array}{c|c} \boldsymbol{b}0 & \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)} \end{array}\right] \right\}, \\
\tilde{\boldsymbol{\theta}}_4^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} \end{array}\right], \left[\begin{array}{c|c} \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} \end{array}\right] \right\}, \qquad i \in \{1, \dots, L-1\}, \\
\tilde{\boldsymbol{\theta}}_4^{(0)} &= \left[\begin{array}{c} \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \\ \hline \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \end{array}\right].
\end{aligned}$$

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}_5^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array}\right], \left[\begin{array}{c|c} \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)} & \boldsymbol{b}0 \end{array}\right] \right\}, \\
\tilde{\boldsymbol{\theta}}_5^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} \end{array}\right], \left[\begin{array}{c|c} \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} \end{array}\right] \right\}, \qquad i \in \{1, \dots, L-1\}, \\
\tilde{\boldsymbol{\theta}}_5^{(0)} &= \left[\begin{array}{c} \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \\ \hline \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \end{array}\right].
\end{aligned}$$

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}_6^{(L)} &= \left\{ \left[\begin{array}{c|c} 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)} & \mathbf{0} \end{array}\right], \left[\begin{array}{c|c} \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)} & \boldsymbol{b}0 \end{array}\right] \right\}, \\
\tilde{\boldsymbol{\theta}}_6^{(i)} &= \left\{ \left[\begin{array}{c|c} 2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{b}0 \end{array}\right], \left[\begin{array}{c|c} \bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{b}0 \end{array}\right] \right\}, \qquad i \in \{1, \dots, L-1\}, \\
\tilde{\boldsymbol{\theta}}_6^{(0)} &= \left[\begin{array}{c} \bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)} \\ \hline \bar{\boldsymbol{\theta}}_{\mathrm{b}}^{(0)} \end{array}\right].
\end{aligned}$$

The arguments to connect $\tilde{\boldsymbol{\theta}}_j$ with $\tilde{\boldsymbol{\theta}}_{j+1}$ are analogous to those previously used to connect $\boldsymbol{\theta}_j$ with $\boldsymbol{\theta}_{j+1}$. We briefly outline them below for completeness.

From $\tilde{\boldsymbol{\theta}}_1$ to $\tilde{\boldsymbol{\theta}}_2$. First, we interpolate from $\boldsymbol{\theta}_{\mathrm{b}}^{(0)}$ to $\bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)}$ with no loss change. Then, for $i = 1, \dots, L-1$, we successively interpolate from $\boldsymbol{b}0$ to $\bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)}$ and from $\boldsymbol{b}0$ to $2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)}$ with no loss change.

From $\tilde{\boldsymbol{\theta}}_5$ to $\tilde{\boldsymbol{\theta}}_6$. For $i = L-1, \dots, 1$, we successively interpolate from $2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ and from $\bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ with no loss change. Finally, we interpolate from $\bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)}$ to $\bar{\boldsymbol{\theta}}_{\mathrm{b}}^{(0)}$ with no loss change.

From $\tilde{\boldsymbol{\theta}}_4$ to $\tilde{\boldsymbol{\theta}}_5$. The output of layer $L-1$ is obtained by stacking the output of two identical sub-networks. Thus, we can interpolate between $\tilde{\boldsymbol{\theta}}_4$ and $\tilde{\boldsymbol{\theta}}_5$ with no change in loss.

From $\tilde{\boldsymbol{\theta}}_3$ to $\tilde{\boldsymbol{\theta}}_4$. For $i = L-1, \dots, 1$, we interpolate from $2\boldsymbol{a}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ and from $\boldsymbol{w}_{\mathrm{t,t}}^{(i)}$ to $\boldsymbol{b}0$ with no change in loss. Then, we interpolate from $\boldsymbol{\theta}_{\mathrm{t}}^{(0)}$ to $\bar{\boldsymbol{\theta}}_{\mathrm{t}}^{(0)}$ with no change in loss. Finally, for $i = 1, \dots, L-1$, we restore sequentially $2\bar{\boldsymbol{a}}_{\mathrm{t,t}}^{(i)}$ and $\bar{\boldsymbol{w}}_{\mathrm{t,t}}^{(i)}$ in the top left corner of the corresponding parameter matrices with no change in loss.

From $\tilde{\boldsymbol{\theta}}_2$ to $\tilde{\boldsymbol{\theta}}_3$. From the previous arguments, we have that $L_N(\tilde{\boldsymbol{\theta}}_2) = L_N(\tilde{\boldsymbol{\theta}}_1) \leq L_N(\boldsymbol{\theta}) + \varepsilon$ and $L_N(\tilde{\boldsymbol{\theta}}_3) = L_N(\tilde{\boldsymbol{\theta}}_6) \leq L_N(\bar{\boldsymbol{\theta}}) + \varepsilon$. First, we interpolate from $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \boldsymbol{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \boldsymbol{0}\ ]\}$ to $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \boldsymbol{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)}\ ]\}$ with no change in loss. Similarly, we interpolate from $\{[\ \boldsymbol{b}0 \mid 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)}\ ], [\ \boldsymbol{b}0 \mid \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)}\ ]\}$ to $\{[\ \boldsymbol{b}0 \mid 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)}\ ]\}$ with no change in loss. Furthermore, as the loss is convex in $\boldsymbol{a}^{(L)}$, we interpolate from $\{[\ 2\boldsymbol{a}_{\mathrm{t}}^{(L)} \mid \boldsymbol{0}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)}\ ]\}$ to $\{[\ \boldsymbol{b}0 \mid 2\bar{\boldsymbol{a}}_{\mathrm{t}}^{(L)}\ ], [\ \boldsymbol{w}_{\mathrm{t}}^{(L)} \mid \bar{\boldsymbol{w}}_{\mathrm{t}}^{(L)}\ ]\}$ while keeping the loss upper bounded by $\max(L_N(\boldsymbol{\theta}), L_N(\bar{\boldsymbol{\theta}})) + \varepsilon$.

□

A.4 Additional Numerical Results

In Figures A.1, A.2 and A.3, we consider the problem of classifying isotropic Gaussians. This is an artificial dataset considered in [MMN18]. The label y is chosen uniformly at random between -1 and 1, i.e., $y \sim \mathrm{Unif}(\{-1, 1\})$. Given y, the feature vector $\boldsymbol{x}$ is a d-dimensional isotropic Gaussian with covariance matrix $(1 + y\Delta)^2 \boldsymbol{I}_d$, i.e., $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{b}0, (1 + y\Delta)^2 \boldsymbol{I}_d)$. We set $d = 32$ and $\Delta = 0.5$, and we run the one-pass (or online) SGD algorithm (3.4) on the two-layer neural network (3.2) with sigmoid activation function ($\sigma(x) = 1/(1 + e^{-x})$). We estimate the population risk and the classification error on 10^4 independent samples. Figure A.1 compares the performance of the trained network (blue dashed curve) and of the dropout network (orange curve) obtained by removing half of the neurons. We plot the population risk and the classification error for $N = 800$ and $N = 6400$. As expected, the performance of the dropout network improves with N, and it is very close to that of the trained network already for $N = 800$. In fact, for $N = 800$ the classification error of the dropout network is $< 0.4\%$. Figure A.2 plots the change in loss between the full and the dropout network, as a function of the number of neurons of the full network N. The change in loss decreases steadily with N for all the values of T taken into account. Finally, Figure A.3 shows that the optimization landscape is approximately connected when $N = 3200$.

In Figures A.4, A.5 and A.6, we consider MNIST classification with a three-layer neural network and CIFAR-10 classification with a four-layer neural network. The results are qualitatively similar to those of Figures 3.1, 3.2 and 3.3 in Section 3.5.

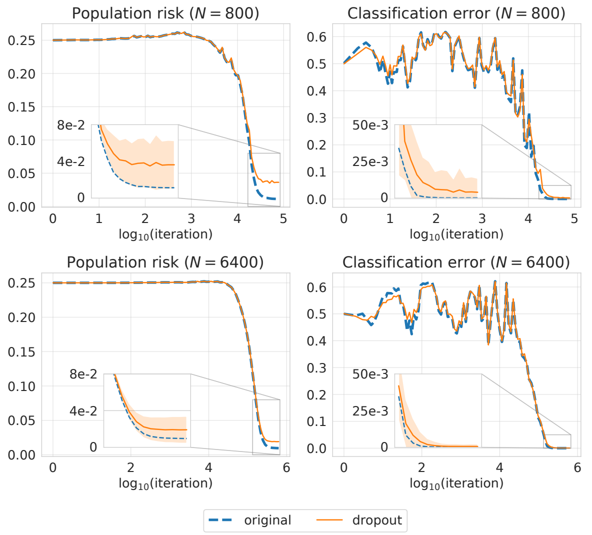


Figure A.1: Comparison of population risk and classification error between the trained network (blue dashed curve) and the dropout network (orange curve) for the classification of isotropic Gaussians.

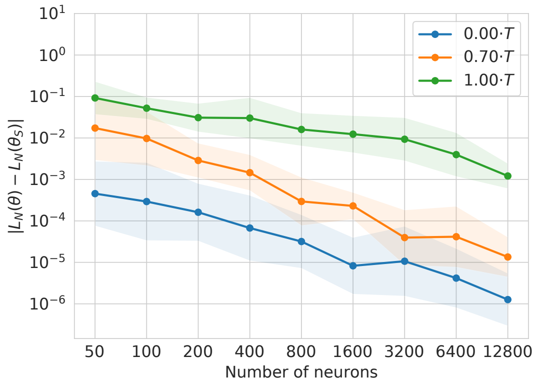


Figure A.2: Change in loss between the full network and the dropout network for the classification of isotropic Gaussians, as a function of the number of neurons N of the full network.

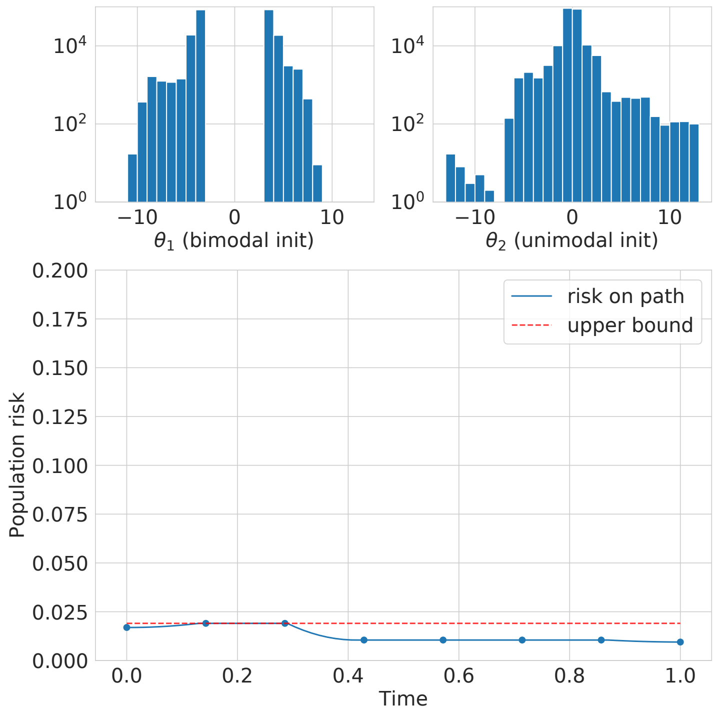
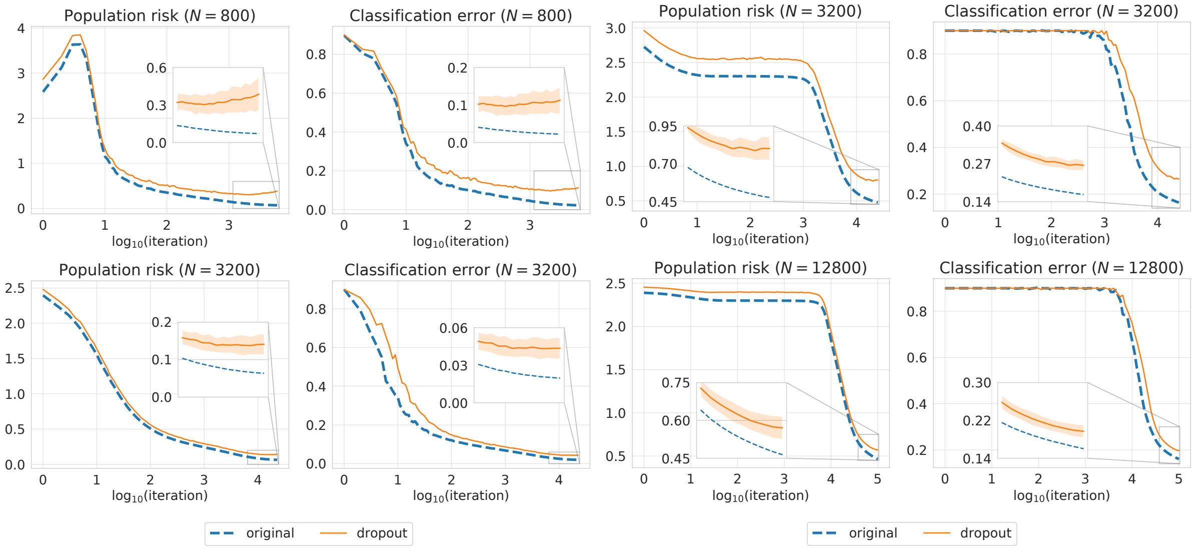


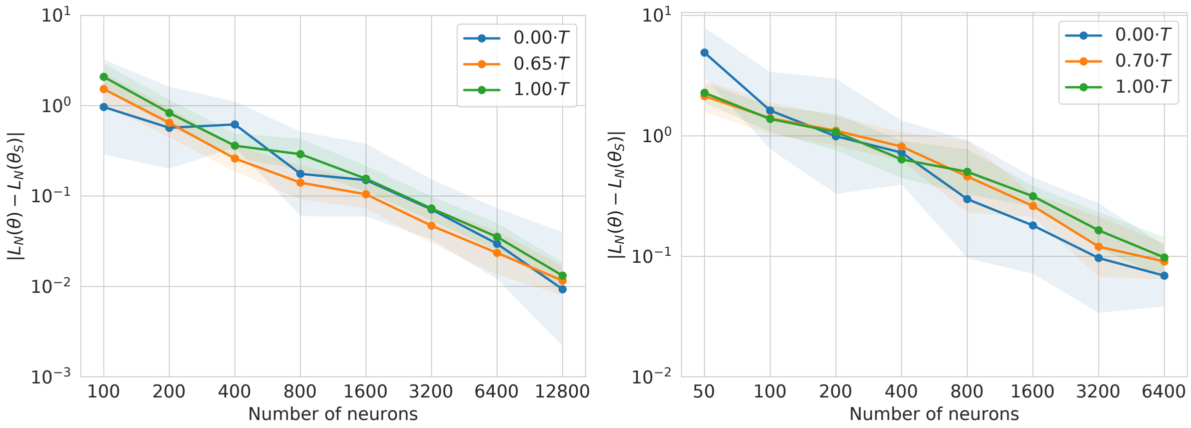
Figure A.3: Classification error along a piecewise linear path that connects two SGD solutions $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ for the classification of isotropic Gaussians with $N = 3200$. The two SGD solutions are initialized with different distributions, and we show their histograms to highlight that $\boldsymbol{\theta}_1$ cannot be obtained by permuting $\boldsymbol{\theta}_2$.



(a) MNIST, three-layer

(b) CIFAR-10, four-layer

Figure A.4: Comparison of population risk and classification error between the trained network (blue dashed curve) and the dropout network (orange curve).



(a) MNIST, three-layer

(b) CIFAR-10, four-layer

Figure A.5: Change in loss after removing half of the neurons from each layer, as a function of the number of neurons N of the full network.

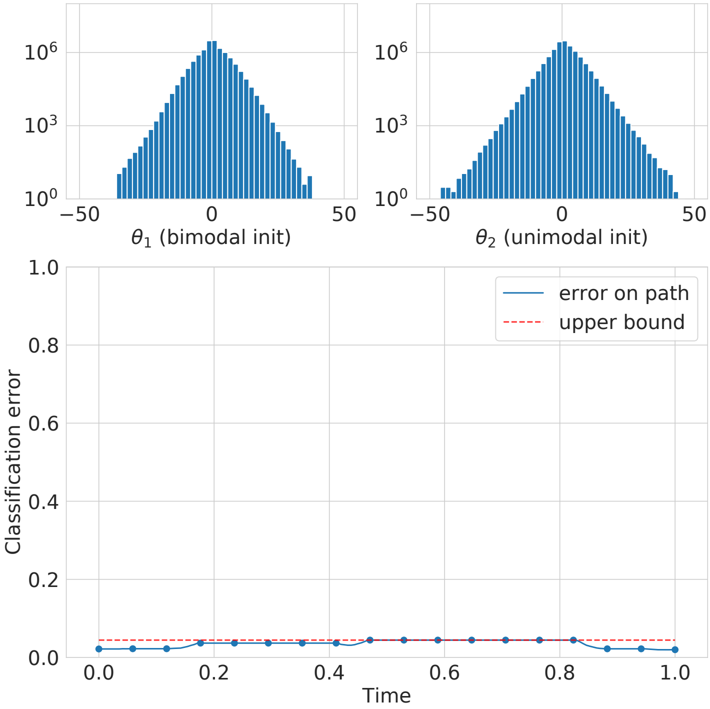


Figure A.6: Classification error along a piecewise linear path that connects two SGD solutions $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ for MNIST classification with a three-layer neural network with $N = 3200$.

APPENDIX B

Appendix for Chapter 4

B.1 Technical Results

In this appendix, we prove a few technical results which are used in the arguments of Section 4.5.2. More specifically, in Section B.1.1 we show that, as $\tau \to \infty$, the minimizer $\rho^*_{\tau,m}(\boldsymbol{\theta})$ of the free energy $\mathcal{F}^{\tau,m}$ converges pointwise in $\boldsymbol{\theta}$ to the minimizer $\rho^*_m(\boldsymbol{\theta})$ of the free energy $\mathcal{F}^m$. This pointwise convergence is needed to establish the result of Lemma 4.5.1. In Section B.1.2, we derive upper bounds on the risk of the minimizer (used in Lemma 4.5.3) and on its second moment (which implies that the sequence of predictors is equi-Lipschitz), and we also prove the lower bound on the partition function in Lemma 4.5.6. Finally, in Section B.1.3 we give the proof of Lemma 4.5.4, which lower bounds the growth of the polynomials f^j and f_j.

B.1.1 Convergence of Minimizers

Lemma B.1.1 (Convergence of densities). *Let $\{\rho_n\}_n$ be a sequence of densities in $\mathcal{K}$ with uniformly bounded truncated entropy, that is*

$$\int \max\{\rho_n(\boldsymbol{\theta})\log\rho_n(\boldsymbol{\theta}), 0\}\,\mathrm{d}\boldsymbol{\theta} \le C, \quad \forall n,$$

for some $C > 0$ that is independent of n, and uniformly bounded second moment, i.e., $M(\rho_n) \le C$ for all n. Then, there exists a subsequence $\{\rho_{n'}\}_{n'}$ of $\{\rho_n\}_n$ and $\rho \in \mathcal{K}$ such that $\rho_{n'} \rightharpoonup \rho$ and

$$C \ge \liminf_{n'\to\infty} M(\rho_{n'}) \ge M(\rho) \ge 0.$$

Proof of Lemma B.1.1. Since $z \mapsto \max\{z\log z, 0\}$, $z \in [0, +\infty)$, has super-linear growth, this result in conjunction with the de la Vallée Poussin criterion (see for instance [HR11]) guarantees that the sequence of densities $\{\rho_n\}_n$ is uniformly integrable. By Dunford-Pettis Theorem (for σ-finite measure spaces, see for instance [Lau15]), relative weak compactness in L_1 is equivalent to uniform integrability. Hence, there exists a density ρ and a subsequence $\{\rho_{n'}\}_{n'}$ of $\{\rho_n\}_n$ such that $\rho_{n'} \rightharpoonup \rho$.

As $M(\cdot)$ is lower-semicontinuous with respect to the topology of weak convergence in L_1 and bounded from below, we have that $\liminf_{n'\to\infty} M(\rho_{n'}) \ge M(\rho)$. Furthermore, as $M(\rho_n) \le C$, we get that $M(\rho) \le C$ and, thus, $\rho \in \mathcal{K}$. □

Lemma B.1.2 (Uniformly bounded $M(\rho^*_{\tau,m})$ and limit of $\rho^*_{\tau,m}$)**.** *Assume that condition **B1** holds. Consider the sequence of minimizing Gibbs distributions $\{\rho^*_{\tau,m}\}_\tau$. The following results hold:*

1. *$M(\rho^*_{\tau,m})$ is uniformly bounded in (τ, m). Moreover, if $\beta\lambda > 1$,*
$$M(\rho^*_m),\ M(\rho^*_{\tau,m}) \le \frac{C_3}{\lambda}, \quad \forall \tau \in (0, +\infty),$$
where $C_3 > 0$ is independent of $(\tau, m, \beta, \lambda)$.

2. *Given any m consistent with **B1**, there exists $\rho_m \in \mathcal{K}$ and a subsequence $\{\rho^*_{\tau',m}\}_{\tau'}$ (which with an abuse of notation we identify with $\{\rho^*_{\tau,m}\}_\tau$) such that $\rho^*_{\tau,m} \rightharpoonup \rho_m$ as $\tau \to \infty$.*

3. *Given any m consistent with **B1**, $\lim_{\tau\to\infty} R_i^{\tau,m}(\rho^*_{\tau,m}) = R_i^m(\rho_m)$ for all $i \in [M]$, and $\liminf_{\tau\to\infty} \mathcal{F}^{\tau,m}(\rho^*_{\tau,m}) \ge \mathcal{F}^m(\rho_m)$.*

Proof of Lemma B.1.2. We provide the proof of the first result for $\rho^*_{\tau,m}$. The arguments for ρ^*_m are the same after changing the notation from $\rho^*_{\tau,m}$ to ρ^*_m. Let $\rho = \mathcal{N}(0, \mathbb{I}_{3\times 3})$. Then, we have that
$$R^{\tau,m}(\rho) = \frac{1}{M}\sum_{i=1}^{M} y_i^2,\ M(\rho) = 3,\ H(\rho) = \frac{3}{2}\ln(2\pi e). \tag{B.1}$$
Note that for this ρ, $R^{\tau,m}(\rho)$, in fact, does not depend on $(\tau, m, \beta, \lambda)$.

From Lemma 10.2 in [MMN18], since $\rho^*_{\tau,m}$ is the unique minimizer of the free energy $\mathcal{F}^{\tau,m}$, we have that the following inequalities hold
$$\mathcal{F}^{\tau,m}(\rho) \ge \mathcal{F}^{\tau,m}(\rho^*_{\tau,m}) \ge R^{\tau,m}(\rho^*_{\tau,m}) + \lambda/4 \cdot M(\rho^*_{\tau,m}) - 1/\beta \cdot [1 + 3 \cdot \log(8\pi/(\beta\lambda))]. \tag{B.2}$$
Furthermore, by using (B.1) and the fact that $\beta > C_1$ and $\lambda < C_2$, we obtain
$$\mathcal{F}^{\tau,m}(\rho) \le K_1 + K_1\lambda - \beta^{-1}K_1 \le K_2, \tag{B.3}$$
for some $K_1, K_2 > 0$ that are independent of $(\tau, m, \beta, \lambda)$. By combining (B.3) and (B.2) and using that $R^{\tau,m}(\rho^*_m) \ge 0$, we conclude that
$$\lambda \cdot M(\rho^*_{\tau,m}) \le K_3 + 1/\beta \cdot [1 + 3 \cdot \log(8\pi/(\beta\lambda))],$$
where $K_3 > 0$ is independent of $(\tau, m, \beta, \lambda)$. As $\beta\lambda > 1$, the first claim immediately follows.

Since the activation and the labels are uniformly bounded in τ and $\{i\}_{i\in[M]}$ is finite, $|R_i^{\tau,m}(\rho^*_{\tau,m})|$ is uniformly bounded in (τ, i). Hence, the following lower bound on the partition function $Z_{\tau,m}(\beta, \lambda)$ holds
$$\begin{aligned}
Z_{\tau,m}(\beta,\lambda) &= \int \exp\left\{-\beta\left[\sum_{i=1}^{M} R_i^{\tau,m}(\rho^*_{\tau,m}) \cdot a^{\tau,m}(w^{\tau,m}x_i + b)^m_\tau + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\} \mathrm{d}\boldsymbol{\theta} \\
&\ge \int \exp\left\{-\beta\left[\sum_{i=1}^{M} |R_i^{\tau,m}(\rho^*_{\tau,m})| \cdot 2m^3 + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\} \mathrm{d}\boldsymbol{\theta} \\
&\ge K_4 \int \exp\left\{-\frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\} \mathrm{d}\boldsymbol{\theta} = \frac{K_5}{\sqrt{\beta^3\lambda^3}} \ge K_6,
\end{aligned} \tag{B.4}$$

for some $K_4, K_5, K_6 > 0$ independent of τ (but dependent on (m, β, λ)). In the same way, one can upper bound $\rho^*_{\tau,m} \cdot Z_{\tau,m}(\beta, \lambda)$ as

$$\exp\left\{-\beta\left[\sum_{i=1}^{M} R_i^{\tau,m}(\rho^*_{\tau,m}) \cdot a^{\tau,m}(w^{\tau,m}x_i + b)^m_\tau + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\} \leq K_7 \exp\left\{-\frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\}, \quad \text{(B.5)}$$

where $K_7 > 0$ is independent of τ (but dependent on (m, β, λ)). Notice that we can increase K_7 to be arbitrarily large and still satisfy (B.5), and in particular, increase it to satisfy $K_7/K_6 > 1$. Thus, by combining (B.4) and (B.5), we get

$$\begin{aligned}
&\int \max\{\rho^*_{\tau,m}(\boldsymbol{\theta}) \ln \rho^*_{\tau,m}(\boldsymbol{\theta}), 0\} \mathrm{d}\boldsymbol{\theta} \\
\leq &\int \max\left\{\frac{K_7}{K_6} \exp\left\{-\frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\} \cdot \left(\ln \frac{K_7}{K_6} - \frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right), 0\right\} \mathrm{d}\boldsymbol{\theta} \\
= &\int_\Omega \frac{K_7}{K_6} \exp\left\{-\frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\} \cdot \left(\ln \frac{K_7}{K_6} - \frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right) \mathrm{d}\boldsymbol{\theta} \leq \int_\Omega \frac{K_7}{K_6} \ln \frac{K_7}{K_6} \mathrm{d}\boldsymbol{\theta},
\end{aligned}$$

where

$$\Omega = \left\{\boldsymbol{\theta} \in \mathbb{R}^3 : \|\boldsymbol{\theta}\|_2^2 \leq \ln\left(\frac{K_7}{K_6}\right) \frac{2}{\beta\lambda}\right\}.$$

Since $\mathrm{vol}(\Omega) < K_8$ for some $K_8 \geq 0$ independent of τ, we get that

$$\int \max\{\rho^*_{\tau,m}(\boldsymbol{\theta}) \ln \rho^*_{\tau,m}(\boldsymbol{\theta}), 0\} \mathrm{d}\boldsymbol{\theta} \leq K_8 \cdot \frac{K_7}{K_6} \cdot \ln \frac{K_7}{K_6},$$

where the RHS is independent of τ. As $M(\rho^*_{\tau,m})$ is uniformly bounded in τ, we can invoke Lemma B.1.1 to finish the proof of the second statement.

We now prove the third statement. By the triangle inequality, we have that, for all $i \in [M]$,

$$\begin{aligned}
&\lim_{\tau\to\infty} \left|\int a^{\tau,m}(w^{\tau,m}x_i + b)^m_\tau \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} - \int a^m(w^m x_i + b)^m_+ \rho_m(\mathrm{d}\boldsymbol{\theta})\right| \\
\leq &\lim_{\tau\to\infty} \left|\int a^{\tau,m}(w^{\tau,m}x_i + b)^m_\tau \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} - \int a^m(w^m x_i + b)^m_+ \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\right| \\
+ &\lim_{\tau\to\infty} \left|\int a^m(w^m x_i + b)^m_+ \rho^*_{\tau,m}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} - \int a^m(w^m x_i + b)^m_+ \rho_m(\mathrm{d}\boldsymbol{\theta})\right| := A_1 + A_2.
\end{aligned}$$

By upper bounding $\rho^*_{\tau,m}$ as in (B.4)-(B.5), we have

$$A_1 \leq K_9 \lim_{\tau\to\infty} \int |a^{\tau,m}(w^{\tau,m}x_i + b)^m_\tau - a^m(w^m x_i + b)^m_+| \exp\left\{-\frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\} \mathrm{d}\boldsymbol{\theta},$$

where $K_9 > 0$ is independent of τ. Thus, an application of the Dominated Convergence theorem gives that the term A_1 vanishes. Furthermore, the term A_2 vanishes by weak convergence of $\rho^*_{\tau,m}$ to ρ_m. This proves that, as $\tau \to \infty$, $y^{\sigma^*}_{\rho^*_{\tau,m}}(x_i) \to y^{\sigma^*}_{\rho_m}(x_i)$ and so $R_i^{\tau,m}(\rho^*_{\tau,m}) \to R_i^m(\rho_m)$.

Note that $-H(\cdot)$ and $M(\cdot)$ are lower-semicontinuous in $\mathcal{K}$. Furthermore, $M(\cdot)$ is lower bounded and $-H(\cdot)$ is lower bounded by Lemma 10.1 in [MMN18] on the subsequence $\{\rho^*_{\tau,m}\}_\tau$, as $M(\rho^*_{\tau,m})$ is uniformly bounded in τ. Hence, as $\rho^*_{\tau,m}$ converges weakly to $\rho_m \in \mathcal{K}$, we conclude that

$$\liminf_{\tau\to\infty} -H(\rho^*_{\tau,m}) \geq -H(\rho_m), \quad \liminf_{\tau\to\infty} M(\rho^*_{\tau,m}) \geq M(\rho_m),$$

which, combined with $R_i^{\tau,m}(\rho^*_{\tau,m}) \to R_i^m(\rho_m)$, implies the desired result. □

Lemma B.1.3 (Pointwise convergence of free-energies)**.** *Fix some distribution $\rho \in \mathcal{K}$, then we have the following pointwise convergence:*

$$\lim_{\tau \to \infty} \mathcal{F}^{\tau,m}(\rho) = \mathcal{F}^m(\rho).$$

Proof of Lemma B.1.3. By construction, we have that $(x)_\tau^m$ converges to $(x)_+^m$, for all $x \in \mathbb{R}$. It is clear that

$$|a^{\tau,m}|(w^{\tau,m}x + b)_\tau^m \rho(\boldsymbol{\theta}) \leq 2m^3 \rho(\boldsymbol{\theta}),$$

and the RHS is integrable. Thus, an application of the Dominated Convergence theorem gives that

$$\lim_{\tau \to \infty} R^{\tau,m}(\rho) = R^m(\rho).$$

This concludes the proof since $M(\rho)$ and $H(\rho)$ are independent of τ. □

Lemma B.1.4 (Pointwise convergence of minimizers)**.** *Assume that condition **B1** holds and consider any satisfactory m. Then, as $\tau \to \infty$, the minimizer $\rho^*_{\tau,m}$ of the free energy $\mathcal{F}^{\tau,m}$ converges pointwise in $\boldsymbol{\theta}$ to the minimizer ρ^*_m of the free energy $\mathcal{F}^m$, i.e.,*

$$\lim_{\tau \to \infty} \rho^*_{\tau,m}(\boldsymbol{\theta}) = \rho^*_m(\boldsymbol{\theta}), \qquad \forall \boldsymbol{\theta} \in \mathbb{R}^3.$$

Proof of Lemma B.1.4. From Lemma B.1.2, we have that there exists a subsequence $\{\rho^*_{\tau,m} \in \mathcal{K}\}$ and $\rho_m \in \mathcal{K}$ such that the following holds

$$\liminf_{\tau \to \infty} \mathcal{F}^{\tau,m}(\rho^*_{\tau,m}) \geq \mathcal{F}^m(\rho_m). \tag{B.6}$$

Since $\rho^*_{\tau,m} \in \mathcal{K}$ minimizes $\mathcal{F}^{\tau,m}$, we have

$$\mathcal{F}^{\tau,m}(\rho^*_{\tau,m}) \leq \mathcal{F}^{\tau,m}(\rho^*_m).$$

By taking the liminf on both sides, using Lemma B.1.3 and (B.6), we have

$$\mathcal{F}^m(\rho_m) \leq \liminf_{\tau \to \infty} \mathcal{F}^{\tau,m}(\rho^*_{\tau,m}) \leq \liminf_{\tau \to \infty} \mathcal{F}^{\tau,m}(\rho^*_m) = \mathcal{F}^m(\rho^*_m).$$

Since ρ^*_m is the unique minimizer of $\mathcal{F}^m$ (see Lemma 10.2 of [MMN18]), ρ^*_m and ρ_m coincide almost everywhere, which implies that

$$R_i^m(\rho_m) = R_i^m(\rho^*_m).$$

Hence, by Lemma B.1.2, we have that

$$\lim_{\tau \to \infty} R_i^{\tau,m}(\rho^*_{\tau,m}) = R_i^m(\rho^*_m).$$

Recall that, by construction, for any parameter $v \in \mathbb{R}$, the τ-smooth m-truncation $v^{\tau,m}$ converges to v^m as $\tau \to \infty$. Furthermore, as $\tau \to \infty$, the smooth m-truncation $(\cdot)_\tau^m$ of the softplus activation converges pointwise to the smooth m-truncation $(\cdot)_+^m$ of the ReLU activation. Thus,

$$\lim_{\tau \to \infty} \Psi_\tau(\boldsymbol{\theta}) = \lim_{\tau \to \infty} \sum_{i=1}^M R_i^{\tau,m}(\rho^*_{\tau,m}) \cdot a^{\tau,m}(w^{\tau,m}x_i + b)_\tau^m = \sum_{i=1}^M R_i^m(\rho^*_m) \cdot a^m(w^m x_i + b)_+^m = \Psi(\boldsymbol{\theta}),$$

where the convergence is intended to be pointwise in $\boldsymbol{\theta}$. Note that $\Psi_\tau(\boldsymbol{\theta})$ is uniformly bounded in τ, hence

$$\lim_{\tau \to \infty} \exp\left\{-\beta \Psi_\tau(\boldsymbol{\theta}) - \frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\} = \exp\left\{-\beta \Psi(\boldsymbol{\theta}) - \frac{\beta\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right\},$$

which implies that $Z_{\tau,m}\rho^*_{\tau,m}(\boldsymbol{\theta})$ converges pointwise to $Z_m \rho^*_m(\boldsymbol{\theta})$. Furthermore, as $\tau \to \infty$, $Z_{\tau,m}$ converges to Z_m by Dominated Convergence, which concludes the proof. □

B.1.2 Bounds on Risk of Minimizer, Second Moment and Partition Function

Lemma B.1.5 (Bound on risk of the minimizer). *Assume that condition **B1** holds. Then,*

$$R^m(\rho_m^*) \le C\lambda,$$

where $C > 0$ is a constant independent of (m, β, λ). In addition, for any $\varepsilon > 0$, there exists $\bar{\tau}(\varepsilon, m, \beta, \lambda)$ such that for any $\tau > \bar{\tau}(\varepsilon, m, \beta, \lambda)$ we have

$$R^{\tau,m}(\rho_{\tau,m}^*) \le C\lambda + \varepsilon.$$

Proof of Lemma B.1.5. Consider a "saw-tooth" function centered at x_i with height y_i and width $\varepsilon > 0$, namely,

$$\mathrm{ST}_{x_i,y_i}(x) := \begin{cases} 0, & x < x_i - \varepsilon \text{ or } x > x_i + \varepsilon, \\ \frac{y_i}{\varepsilon}(x - x_i + \varepsilon), & x_i - \varepsilon \le x \le x_i, \\ \frac{y_i}{\varepsilon}(x_i - x + \varepsilon), & x_i < x \le x_i + \varepsilon, \end{cases}$$

Notice that this function can be implemented by the following $\hat{\rho}_i$:

$$\hat{\rho}_i = \frac{1}{3}\left(\delta_{\left(\frac{3y_i}{\varepsilon},1,\varepsilon - x_i\right)} + \delta_{\left(-\frac{6y_i}{\varepsilon},1,-x_i\right)} + \delta_{\left(\frac{3y_i}{\varepsilon},1,-\varepsilon - x_i\right)}\right),$$

in the sense that

$$\mathrm{ST}_{x_i,y_i}(x) = \int a(wx+b)_+ \hat{\rho}_i(\mathrm{d}\boldsymbol{\theta}),$$

where $\delta_{\boldsymbol{\theta}}$ stands for a delta distribution centered at the point $\boldsymbol{\theta} = (a, w, b) \in \mathbb{R}^3$. Let us pick ε such that $\varepsilon < \min_{i \in [M-1]} \{|x_i - x_{i+1}|/2\}$. This condition on ε guarantees that

$$\left\{x \in \mathbb{R} : \int a(wx+b)_+ \hat{\rho}_i(\mathrm{d}\boldsymbol{\theta}) \neq 0\right\} \cap \left\{x \in \mathbb{R} : \int a(wx+b)_+ \hat{\rho}_j(\mathrm{d}\boldsymbol{\theta}) \neq 0\right\} = \emptyset, \ \forall i \neq j,$$

which ensures that the "saw-tooth" functions are not intersecting. Define

$$\hat{\rho} = \frac{1}{3M} \sum_{i=1}^{M} \left[\delta_{\left(\frac{3My_i}{\varepsilon},1,\varepsilon - x_i\right)} + \delta_{\left(-\frac{6My_i}{\varepsilon},1,-x_i\right)} + \delta_{\left(\frac{3My_i}{\varepsilon},1,-\varepsilon - x_i\right)}\right].$$

Then, one immediately has that, for all $i \in [M]$,

$$\int a(wx_i + b)_+ \hat{\rho}(\mathrm{d}\boldsymbol{\theta}) = y_i.$$

Furthermore, by taking a sufficiently large m, in particular, taking $m > \max_i\{6M|y_i|/\varepsilon\} + 3|x_M| + 3|x_1| + 2$ suffices, we get that, for all $x \in [x_1, x_M]$,

$$\int a^m(w^m x + b)_+^m \hat{\rho}(\mathrm{d}\boldsymbol{\theta}) = \int a(wx+b)_+ \hat{\rho}(\mathrm{d}\boldsymbol{\theta}),$$

which implies that $R^m(\hat{\rho}) = 0$.

Let $\mathcal{N}(\mu, \sigma^2)$ denote a Gaussian distribution with mean $\mu \in \mathbb{R}$ and variance $\sigma^2 \in \mathbb{R}$, and let $U(\mu, \sigma^2)$ denote the uniform distribution with mean μ and variance $\sigma^2/12$. Given $(\mu_1, \mu_2, \mu_3) \in \mathbb{R}^3$ and $\sigma^2 \in \mathbb{R}$, let $\rho_{((\mu_1,\mu_2,\mu_3),\sigma^2)}$ denote the following product distribution

$$\rho_{((\mu_1,\mu_2,\mu_3),\sigma^2)} := U(\mu_1, \sigma^2) \times \mathcal{N}(\mu_2, \sigma^2) \times \mathcal{N}(\mu_3, \sigma^2),$$

and define

$$\tilde{\rho} = \frac{1}{3M} \sum_{i=1}^{M} \left[\rho_{((\frac{3My_i}{\varepsilon}, 1, \varepsilon - x_i), \sigma^2)} + \rho_{((-\frac{6My_i}{\varepsilon}, 1, -x_i), \sigma^2)} + \rho_{((\frac{3My_i}{\varepsilon}, 1, -\varepsilon - x_i), \sigma^2)} \right]. \tag{B.7}$$

Note that, for $\sigma^2 < 1$ and m chosen sufficiently large as mentioned previously,

$$\int a^m (w^m x + b)_+^m \tilde{\rho}(\mathrm{d}\boldsymbol{\theta}) = \int a (w^m x + b)_+^m \tilde{\rho}(\mathrm{d}\boldsymbol{\theta}).$$

Thus, by computing the integral w.r.t. a, we have that

$$\begin{aligned}
&\int a^m (w^m x + b)_+^m \hat{\rho}(\mathrm{d}\boldsymbol{\theta}) - \int a^m (w^m x + b)_+^m \tilde{\rho}(\mathrm{d}\boldsymbol{\theta}) \\
&= \sum_{i=1}^{M} \left[\frac{y_i}{\varepsilon} \left(\int (w^m x + b)_+^m \delta_{(1, \varepsilon - x_i)}(\mathrm{d}w\, \mathrm{d}b) - \int (w^m x + b)_+^m \rho_{((1, \varepsilon - x_i), \sigma^2)}(\mathrm{d}w\, \mathrm{d}b) \right) \right] \\
&- \sum_{i=1}^{M} \left[\frac{2y_i}{\varepsilon} \left(\int (w^m x + b)_+^m \delta_{(1, -x_i)}(\mathrm{d}w\, \mathrm{d}b) - \int (w^m x + b)_+^m \rho_{((1, -x_i), \sigma^2)}(\mathrm{d}w\, \mathrm{d}b) \right) \right] \\
&+ \sum_{i=1}^{M} \left[\frac{y_i}{\varepsilon} \left(\int (w^m x + b)_+^m \delta_{(1, -\varepsilon - x_i)}(\mathrm{d}w\, \mathrm{d}b) - \int (w^m x + b)_+^m \rho_{((1, -\varepsilon - x_i), \sigma^2)}(\mathrm{d}w\, \mathrm{d}b) \right) \right],
\end{aligned} \tag{B.8}$$

where, with an abuse of notation, we denote by $\rho_{((\mu_2, \mu_3), \sigma^2)}$ the marginal of $\rho_{((\mu_1, \mu_2, \mu_3), \sigma^2)}$ with respect to the last two components. By applying to Kantorovich-Rubinstein theorem (see, for instance, [Vil09]), we have that

$$K \cdot W_1(p, q) = \sup_{\|f\|_{\mathrm{Lip}} \le K} |\mathbb{E}_{x \sim p} f(x) - \mathbb{E}_{y \sim q} f(y)|, \tag{B.9}$$

for two densities p and q, where W_1 is the 1-Wasserstein distance and $\|f\|_{\mathrm{Lip}}$ denotes the Lipschitz constant of f. Notice that $(w^m x + b)_+^m$ is Lipschitz in (w, b) with Lipschitz constant upper bounded by $\max(|x|, 1)$. Hence, combining (B.8) and (B.9), we have that

$$\begin{aligned}
&\left(\int a^m (w^m x + b)_+^m \hat{\rho}(\mathrm{d}\boldsymbol{\theta}) - \int a^m (w^m x + b)_+^m \tilde{\rho}(\mathrm{d}\boldsymbol{\theta}) \right)^2 \\
&\quad \le K_1 \Bigg(\sum_{i=1}^{M} W_1(\delta_{(1, \varepsilon - x_i)}, \rho_{((1, \varepsilon - x_i), \sigma^2)}) + W_1(\delta_{(1, -x_i)}, \rho_{((1, -x_i), \sigma^2)}) \\
&\qquad + W_1(\delta_{(1, -\varepsilon - x_i)}, \rho_{((1, -\varepsilon - x_i), \sigma^2)}) \Bigg)^2,
\end{aligned} \tag{B.10}$$

where $K_1 > 0$ is a constant independent of m. Recalling the form of the 2-Wasserstein distance between a delta and a Gaussian distribution, we have that

$$W_2^2(\delta_{(w,b)}, \rho_{((w,b), \sigma^2)}) \le K_2 \sigma^2, \tag{B.11}$$

for some constant $K_2 > 0$. As the W_1 distance is upper bounded by the W_2 distance (via Hölder's inequality), by combining (B.10) and (B.11), we conclude that

$$\left(\int a^m (w^m x + b)_+^m \hat{\rho}(\mathrm{d}\boldsymbol{\theta}) - \int a^m (w^m x + b)_+^m \tilde{\rho}(\mathrm{d}\boldsymbol{\theta}) \right)^2 \le K_3 \sigma^2,$$

where $K_3 > 0$ is a constant independent of m. Hence, by taking $\sigma^2 = \min(\lambda, 1/2)$, we have

$$R^m(\tilde{\rho}) \leq K_4\lambda,$$

where $K_4 > 0$ is a constant independent of m.

Now recall that the differential entropy is a concave function of the distribution. Hence, by using the fact that $\rho_{((\mu_1,\mu_2,\mu_3),\sigma^2)}$ is a product distribution and by explicitly computing the entropy of a Gaussian and a uniform random variable, we conclude that

$$H(\tilde{\rho}) \geq K_5(-1 + \log\lambda),$$

where $K_5 > 0$ is a constant independent of m. As $M(\tilde{\rho})$ is upper bounded by a constant independent of m, we conclude that

$$\mathcal{F}^m(\tilde{\rho}) \leq K_6\lambda + \frac{K_5}{\beta}(1 - \log\lambda), \tag{B.12}$$

with $K_6 > 0$ independent of m. Hence, since ρ^*_m is the minimizer of the free energy, by using the bound from Lemma 10.2 in [MMN18], we get that

$$\frac{1}{2}R^m(\rho^*_m) \leq K_6\lambda + \frac{K_5}{\beta}(1 - \log\lambda) + \frac{1}{\beta}\left[1 + 3\log\frac{8\pi}{\beta\lambda}\right]. \tag{B.13}$$

Since $\beta > -\frac{1}{\lambda}\log\lambda$ and $\beta\lambda > 1$, (B.13) implies that

$$R^m(\rho^*_m) \leq K_7\lambda, \tag{B.14}$$

for $K_7 > 0$ independent of (m, β, λ). This finishes the proof of the first part of the statement. The second part of the statement follows by combining (B.14) with Lemma B.1.4. □

Lemma B.1.6 (Second moment is uniformly bounded)**.** *Assume that condition **B1** holds. It holds that there exists $\tau(m, \beta, \lambda)$ such that for any $\tau > \tau(m, \beta, \lambda)$ the following upper bound holds:*

$$M(\rho^*_{\tau,m}) \leq C,$$

for some $C > 0$ that is independent of $(\tau, m, \beta, \lambda)$.

Proof of Lemma B.1.6. Let $\tilde{\rho}$ be defined as in (B.7). Then, by combining (B.12) with Lemma B.1.3, we have that, for $\tau > \tau(m, \beta, \lambda)$,

$$\mathcal{F}^{\tau,m}(\tilde{\rho}) \leq K_1\lambda + \frac{K_2}{\beta}(1 - \log\lambda),$$

where $K_1, K_2 > 0$ are independent of m. Hence, by using (B.2) with $\tilde{\rho}$ in place of ρ and by recalling that $R^{\tau,m}(\rho^*_{\tau,m}) \geq 0$ and the existence of constants C_1 and C_2 such that $\beta > C_1$ and $\lambda < C_2$, the result readily follows. □

We conclude this part of the appendix by providing the proof of Lemma 4.5.6.

Proof of Lemma 4.5.6. Consider the following lower bound

$$\begin{aligned}
Z_m(\lambda,\beta) &\geq \int \exp\left\{-\beta\left[\sum_{i=1}^{M}|R_i^m(\rho_m^*)|\cdot|a^m|(w^m x_i+b)_+^m+\frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\}\mathrm{d}\boldsymbol{\theta}\\
&\geq \int \exp\left\{-\beta\left[\sum_{i=1}^{M}|R_i^m(\rho_m^*)|\cdot|a|(w^m x_i+b)_+ +\frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\}\mathrm{d}\boldsymbol{\theta}\\
&\geq \int \exp\left\{-\beta\left[\sum_{i=1}^{M}|x_iR_i^m(\rho_m^*)|\cdot|aw|+\sum_{i=1}^{M}|R_i^m(\rho_m^*)|\cdot|ab|+\frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2\right]\right\}\mathrm{d}\boldsymbol{\theta}.
\end{aligned} \tag{B.15}$$

Define $A=\sum_{i=1}^{M}|x_iR_i^m(\rho_m^*)|$ and $B=\sum_{i=1}^{M}|R_i^m(\rho_m^*)|$. By Lemma B.1.5, $|R_i^m(\rho_m^*)|\leq K_1\sqrt{\lambda}$, where $K_1>0$ is independent of (m,β,λ,i). Therefore, $|A|,|B|\leq K_2\sqrt{\lambda}$ for some $K_2>0$ independent of (m,β,λ). Using the inequalities $2|aw|\leq a^2+w^2$ and $2|ab|\leq a^2+b^2$, the RHS of (B.15) can be lower bounded by

$$\int_{\mathbb{R}^3}\exp\left\{-\frac{\beta}{2}\left[(2K_2\sqrt{\lambda}+\lambda)\cdot a^2+(K_2\sqrt{\lambda}+\lambda)\cdot w^2+(K_2\sqrt{\lambda}+\lambda)\cdot b^2\right]\right\}d\boldsymbol{\theta}.$$

By explicitly computing the integral above, the desired result immediately follows. □

B.1.3 Lower Bound on Polynomials

Proof of Lemma 4.5.4. We start by rewriting P_2 as

$$\begin{aligned}
P_2(x) &= (1-a^2)\cdot(x-x_c)^2+\Big[2(1-a^2)x_c+2ab\Big]\cdot(x-x_c)\\
&\qquad\qquad\qquad\qquad +(1-a^2)x_c^2+2abx_c+1-b^2\\
&= \frac{1}{2}P_2''(x_c)\cdot(x-x_c)^2+P_2'(x_c)\cdot(x-x_c)+P_2(x_c).
\end{aligned} \tag{B.16}$$

By definition of x_c, one can immediately verify that $P_2(x_c)\geq 0$. Notice that, if $P_2''(x_c)$ is close to 0, then a^2 is close to 1, which implies that (since $x_c\in I$ and, thus, bounded in absolute value) $|P_2'(x_c)|$ is close to $2|b|$ and $P_2(x_c)$ is close to $\operatorname{sign}(a)\cdot 2bx_c+1-b^2$. Therefore, at least one of the coefficients $P_2(x_c),|P_2'(x_c)|,|P_2''(x_c)|$ is lower bounded by a constant that is independent of (a,b).

Next, we distinguish two cases depending on the sign of $P_2''(x_c)$. First, assume that $P_2''(x_c)\geq 0$. We now show that $P_2'(x_c)\cdot(x-x_c)\geq 0$.

In case of a degenerate polynomial, i.e., $P_2''(x_c)=0$, we distinguish two sub-cases: either $P_2'(x_c)>0$ or $P_2'(x_c)<0$ holds. (The case corresponding to $P_2'(x_c)=0$ is trivial.) If $P_2'(x_c)>0$, then by definition of x_c and the fact that $x\in\Omega_+$ by assumption, we have that $(x-x_c)>0$. In fact, recalling the definition of x_r in Definition 4.4.1, as Ω_+ has non-zero Lebesgue measure, x_c is either the left extreme of I (i.e., $x_c=\inf_{\tilde{x}\in I}\tilde{x}$) or $x_c=x_r\in I$ (i.e., in the interior) and, hence, $\Omega_+=(x_r,I_r]$ with $x_r<I_r$. This gives that $P_2'(x_c)(x-x_c)\geq 0$. The case $P_2'(x_c)<0$ follows from similar arguments.

Now assume that $P_2''(x_c)>0$ and let $x_{\min}$ be the minimizer of P_2 on the interval I. If $x\geq x_{\min}$ then, by definition of a critical point, $x_c\geq x_{\min}$ which means that x_c is located on the *right* branch of the parabola and, hence, $P_2'(x_c)\geq 0$. Furthermore, x belongs to the interval $[x_c, I_r]$ by definition of x_c. These facts imply that $P_2'(x_c) \cdot (x - x_c) \geq 0$. The case $x < x_{\min}$ is treated in a similar fashion.

As it was shown, at least one of the coefficients $P_2(x_c), |P_2'(x_c)|, P_2''(x_c)$ is lower bounded by a constant that is independent of (a, b), and $P_2'(x_c)(x - x_c) \geq 0$, hence, choosing

$$\alpha_2 = P_2''(x_c), \quad \alpha_1 = |P_2'(x_c)|, \quad \alpha_0 = P_2(x_c),$$

concludes the proof for the case of non-negative curvature.

Assume now that $P_2''(x_c) < 0$. As $|\Omega_+|$ is lower bounded by a strictly positive constant, we can pick $\tilde{x} \in \Omega_+$ such that $|\tilde{x} - x_c| = C$, for some $C > 0$ which is independent of (a, b). As $\tilde{x} \in \Omega_+$, we have that $P_2(\tilde{x}) \geq 0$. Furthermore, by rewriting $P_2(\tilde{x})$ as in (B.16), we obtain that

$$\frac{1}{2}P_2''(x_c) \cdot (\tilde{x} - x_c)^2 + P_2'(x_c) \cdot (\tilde{x} - x_c) + P_2(x_c) \geq 0,$$

which implies that

$$|P_2'(x_c)||\tilde{x} - x_c| + P_2(x_c) \geq -\frac{1}{2}P_2''(x_c)(\tilde{x} - x_c)^2. \tag{B.17}$$

As $|\tilde{x} - x_c| = C$, (B.17) is equivalent to

$$|P_2'(x_c)| \cdot C + P_2(x_c) \geq -\frac{1}{2}P_2''(x_c) \cdot C^2. \tag{B.18}$$

Now, if both $|P_2'(x_c)|$ and $P_2(x_c)$ are close to 0, then (B.18) immediately implies that $P_2''(x_c)$ is also close to 0. However, following the argument above, it is not possible that $-P_2''(x_c)$, $|P_2'(x_c)|$ and $P_2(x_c)$ are simultaneously close to 0. This proves that $\max(|P_2'(x_c)|, P_2(x_c))$ is lower bounded by a constant that is independent of (a, b).

Let $x_{\max}$ be the maximizer of P_2 and, without loss of generality, assume that $x_c < x_{\max}$ (the case $x_c \geq x_{\max}$ is handled in a similar way). Note that, by definition of x_c, the point x lies in the interval $[x_c, x_{\max}]$. To show this, let us assume the contrary, i.e., $x > x_{\max}$ (the case $x < x_c < x_{\max}$ is ruled out by the assumption that $x \in \Omega_+$). Then, the root of P_2 which is the closest in Euclidean distance to x is located to the right of $x_{\max}$, hence $x_c < x_{\max}$ cannot be a critical point for x, which leads to a contradiction. This proves that x lies in the interval $[x_c, x_{\max}]$ and in particular, $x \leq x_{\max}$. Furthermore, by concavity, the parabola $P_2(\tilde{x})$ is lower bounded by the line that connects $(x_c, P_2(x_c))$ and $(x_{\max}, P_2(x_{\max}))$ for $\tilde{x} \in [x_c, x_{\max}]$. By the focal property of the parabola, this line has angular coefficient $|P_2'(x_c)|/2$. Therefore,

$$P_2(\tilde{x}) \geq (\tilde{x} - x_c) \cdot |P_2'(x_c)|/2 + P_2(x_c), \quad \tilde{x} \in [x_c, x_{\max}].$$

Picking $\tilde{x} = x$ and

$$\alpha_2 = 0, \quad \alpha_1 = |P_2'(x_c)|/2, \quad \alpha_0 = P_2(x_c),$$

gives the desired result in the case $P_2''(x_c) < 0$ and concludes the proof. □

APPENDIX C

Appendix for Chapter 5

C.1 Closed Forms for the Population Risk

For the proofs of Lemmas 5.4.1 and 5.5.1 in the current section, we assume that the rows of $\boldsymbol{B}$ have non-zero norm, hence, in particular, they may be chosen to have unit norm. In the end of the section, we elaborate on why this assumption holds true.

Let us also mention that we call σ odd in L^2 sense. For this particular case, it means that $\sigma(x) = \sigma(-x)$ for $x \neq 0$ and $|\sigma(0)| < C$, where C is some universal constant. This concern is purely technical, since the main application of our results is *1-bit* compression. Namely, we do not set $\sigma(0) = \operatorname{sgn}(0) = 0$. In fact, this would mean that the compressed sequence can take values in $\{-1, 0, 1\}$, which would not result in *1-bit* compression, but rather in $\log_2(3)$-*bits* compression. It is safe to ignore this technicality and intuitively assume that $\sigma(0) = 0$.

Proof of Lemma 5.4.1. Opening up the two-norm gives

$$\mathbb{E}\|\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 = \mathbb{E}\|\boldsymbol{x}\|_2^2 + \mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 - 2\mathbb{E}\langle \boldsymbol{x}, \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\rangle. \tag{C.1}$$

Since $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, we get

$$\mathbb{E}\|\boldsymbol{x}\|_2^2 = d. \tag{C.2}$$

Let $\boldsymbol{B}^\top = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_n] \in \mathbb{R}^{d\times n}$ and $\boldsymbol{A} = [\boldsymbol{a}_1, \dots, \boldsymbol{a}_n] \in \mathbb{R}^{d\times n}$, with $\|\boldsymbol{b}_i\|_2 = \|\boldsymbol{B}_{i,:}\| = 1$. Rewriting the second term in (C.1) gives

$$\mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 = \sum_{i,j=1}^{n} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(\langle \boldsymbol{b}_i, \boldsymbol{x}\rangle) \cdot \sigma(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle)\right]. \tag{C.3}$$

Using the reproducing property of Hermite coefficients (see, e.g., Chapter 11 in [O'D14]), since the random variables $\langle \boldsymbol{b}_i, \boldsymbol{x}\rangle$ and $\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle$ are $\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle$-correlated, we have

$$\mathbb{E}\left[h_{2\ell+1}(\langle \boldsymbol{b}_i, \boldsymbol{x}\rangle) \cdot h_{2\ell+1}(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle)\right] = \langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle^{2\ell+1}, \quad \mathbb{E}\left[h_{2\ell+1}(\langle \boldsymbol{b}_i, \boldsymbol{x}\rangle) \cdot h_{2k+1}(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle)\right] = 0,$$

for $k \neq \ell$. This gives that

$$\mathbb{E}\left[\sigma(\langle \boldsymbol{b}_i, \boldsymbol{x}\rangle) \cdot \sigma(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle)\right] = \sum_{\ell=0}^{\infty} (c_{2\ell+1})^2 \langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle^{2\ell+1} = f(\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle),$$

and, hence, using (C.3) we arrive to

$$\mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 = \sum_{i,j=1}^{n} \langle \boldsymbol{a}_i, \boldsymbol{a}_j \rangle \cdot f(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle) = \mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top)\right]. \tag{C.4}$$

Rearranging the last term in (C.1) gives

$$\mathbb{E}\langle \boldsymbol{x}, \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\rangle = \sum_{i=1}^{d}\sum_{j=1}^{n} a_j^i \cdot \mathbb{E}[x_i \sigma(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle)], \tag{C.5}$$

where a_j^i stands for the i-th coordinate of the vector $\boldsymbol{a}_j$ and x_i stands for the i-th coordinate of the vector $\boldsymbol{x}$. Let us now compute the inner expected value for each pair (i, j). Notice that the random variables $\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle$ and x_i are jointly Gaussian with zero mean and covariance matrix $\widetilde{\boldsymbol{\Sigma}} \in \mathbb{R}^{2\times 2}$:

$$\widetilde{\boldsymbol{\Sigma}}_{21} = \widetilde{\boldsymbol{\Sigma}}_{12} = \mathbb{E}x_i\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle = \mathbb{E}b_j^i x_i^2 = b_j^i, \quad \widetilde{\boldsymbol{\Sigma}}_{11} = \mathbb{E}\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle^2 = \|\boldsymbol{b}_j\|_2^2 = 1, \quad \widetilde{\boldsymbol{\Sigma}}_{22} = \mathbb{E}x_i^2 = 1.$$

Hence, the random vectors $(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle, x_i)$ and

$$\left(y_1, b_j^i \cdot y_1 + \sqrt{1-(b_j^i)^2} \cdot y_2\right), \qquad \text{with } (y_1, y_2) \sim \mathcal{N}(0, \boldsymbol{I})$$

are identically distributed. In this view, we obtain

$$\begin{aligned} \mathbb{E}[x_i\sigma(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle)] &= \mathbb{E}\left[\left(b_j^i \cdot y_1 + \sqrt{1-(b_j^i)^2} \cdot y_2\right)\sigma(y_1)\right] \\ &= b_j^i \cdot \mathbb{E}[y_1\sigma(y_1)] + \sqrt{1-(b_j^i)^2} \cdot \mathbb{E}[y_2] \cdot \mathbb{E}[\sigma(y_1)] = c_1 \cdot b_j^i, \end{aligned} \tag{C.6}$$

where we applied the reproducing property to conclude that $\mathbb{E}[y_1\sigma(y_1)] = c_1$. Consequently, by combining (C.5) and (C.6), we get that

$$\mathbb{E}\langle \boldsymbol{x}, \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\rangle = c_1 \cdot \sum_{i=1}^{d}\sum_{j=1}^{n} a_j^i b_j^i = c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right]. \tag{C.7}$$

By combining (C.1), (C.2), (C.4) and (C.7), we obtain the desired expression for $\widetilde{R}(r)$.

Assume now that σ is homogeneous. Then, in (C.3) and (C.5), the norm of $\boldsymbol{b}_i$ can be pushed into the corresponding $\boldsymbol{a}_i$ and, hence, we obtain

$$\min_{\boldsymbol{A},\boldsymbol{B}} \mathbb{E}\|\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 = \min_{\boldsymbol{A},\|\boldsymbol{B}_i\|_2=1} \mathbb{E}\|\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2,$$

which proves that $\widehat{\mathcal{R}}(r) = \widetilde{\mathcal{R}}(r)$.

Finally, consider the case $\sigma(x) = \mathrm{sign}(x)$. Then, Grothendieck's identity (see, e.g., Lemma 3.6.6 in [Ver18]) gives

$$\mathbb{E}\sigma(\langle \boldsymbol{b}_i, \boldsymbol{x}\rangle)\sigma(\langle \boldsymbol{b}_j, \boldsymbol{x}\rangle) = \frac{2}{\pi}\arcsin(\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle) \Rightarrow f(x) = \frac{2}{\pi}\arcsin(x).$$

Recalling that the first Hermite coefficient of $\sigma(x) = \mathrm{sign}(x)$ is equal to $\sqrt{\frac{2}{\pi}}$ finishes the proof. □

Proof of Lemma 5.5.1. The proof of Lemma 5.5.1 follows from similar arguments as that of Lemma 5.4.1. Given this, we only explain the key differences. We first show that it is enough to consider $\boldsymbol{\Sigma} = \boldsymbol{D}^2$. Given the SVD $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{U}^\top$, we have $\boldsymbol{x} = \boldsymbol{U}\boldsymbol{D}\tilde{\boldsymbol{x}}$, where $\tilde{\boldsymbol{x}} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Now, we can push the rotation $\boldsymbol{U}$ in $\boldsymbol{A}, \boldsymbol{B}$:

$$\left\|\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\right\|_2 = \left\|\boldsymbol{D}\tilde{\boldsymbol{x}} - \boldsymbol{U}^\top\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{U}\boldsymbol{D}\tilde{\boldsymbol{x}})\right\|_2.$$

Thus, after replacing $\boldsymbol{A}$ with $\boldsymbol{U}^\top\boldsymbol{A}$ and $\boldsymbol{B}$ with $\boldsymbol{B}\boldsymbol{U}$, we may assume that $\boldsymbol{x} = \boldsymbol{D}\tilde{\boldsymbol{x}}$.

We again open up the two-norm

$$\mathbb{E}\|\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 = \mathbb{E}\|\boldsymbol{x}\|_2^2 + \mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 - 2\mathbb{E}\langle\boldsymbol{x}, \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\rangle. \tag{C.8}$$

For the first term, we clearly have

$$\mathbb{E}\|\boldsymbol{x}\|_2^2 = \mathrm{Tr}\left[\boldsymbol{D}^2\right].$$

Now, for the second term we write

$$\mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 = \mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\|_2^2,$$

where $\tilde{\boldsymbol{x}} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Thus, as in the proof of Lemma 5.4.1, we have

$$\mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\|_2^2 = \mathrm{Tr}\left[\boldsymbol{A}^\top\boldsymbol{A} \cdot f(\boldsymbol{B}\boldsymbol{D}^2\boldsymbol{B}^\top)\right].$$

Similarly, for the last term we obtain

$$\mathbb{E}\langle\boldsymbol{x}, \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\rangle = \mathbb{E}\langle\tilde{\boldsymbol{x}}, \boldsymbol{D}\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\rangle = c_1\mathrm{Tr}\left[\boldsymbol{D}\boldsymbol{A}\boldsymbol{B}\boldsymbol{D}\right].$$

Finally, since σ is homogeneous, by abuse of notation we can replace $\boldsymbol{B}\boldsymbol{D}$ by any $\boldsymbol{B}$ with unit-norm rows. This follows from the fact that, similarly to the proof of Lemma 5.4.1 (namely, equations (C.3) and (C.5)), we have that

$$\begin{aligned}\mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\|_2^2 &= \sum_{i,j=1}^{n}\langle\boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{i,:}, \tilde{\boldsymbol{x}}\rangle) \cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)\right],\\ \mathbb{E}\langle\tilde{\boldsymbol{x}}, \boldsymbol{D}\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\rangle &= \sum_{i=1}^{d}\sum_{j=1}^{n} a_j^i \cdot \mathbb{E}[(D_{i,i} \cdot \tilde{x}_i) \cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)],\end{aligned} \tag{C.9}$$

which, by homogeneity, readily gives that the norm of $(\boldsymbol{B}\boldsymbol{D})_{i,:}$ can be pushed into the corresponding $\boldsymbol{a}_i$.

As a result, the statement of Lemma 5.5.1 readily follows by comparing the terms. □

Rows of $\boldsymbol{B}$ are non-zero. We show that the assumption holds true by contradiction. Without loss of generality, assume that the first $n' \leq n$ rows of $\boldsymbol{B}$ are zero vectors. Hence, from (C.9) we can see that the following holds:

$$\begin{aligned}\mathbb{E}\|\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\|_2^2 = &\sum_{i,j=n'+1}^{n} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{i,:}, \tilde{\boldsymbol{x}}\rangle)\cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)\right] \\ &+ \sum_{i\le n' \ \wedge \ j>n'} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(0)\cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)\right] \\ &+ \sum_{i> n' \ \wedge \ j\le n'} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(0)\cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{i,:}, \tilde{\boldsymbol{x}}\rangle)\right] + \sum_{i,j\le n'} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \sigma(0)^2 \\ = &\sum_{i,j=n'+1}^{n} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{i,:}, \tilde{\boldsymbol{x}}\rangle)\cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)\right] \\ &+ \sum_{i,j\le n'} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \sigma(0)^2 \\ \ge &\sum_{i,j=n'+1}^{n} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \mathbb{E}\left[\sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{i,:}, \tilde{\boldsymbol{x}}\rangle)\cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)\right], \end{aligned} \tag{C.10}$$

where in the fourth line we used that for $\tilde{\boldsymbol{x}} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, as σ is odd, the following identity holds:

$$\mathbb{E}\left[\sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)\right] = 0,$$

and the last inequality follows from the fact that for the Gram matrix $\boldsymbol{M}$ of the vectors $\{\boldsymbol{a}_i\}_{i=1}^{n'}$:

$$\sum_{i,j\le n'} \langle \boldsymbol{a}_i, \boldsymbol{a}_j\rangle \cdot \sigma(0)^2 = \sigma(0)^2 \cdot \langle \boldsymbol{1}, \boldsymbol{M}\boldsymbol{1}\rangle \ge 0.$$

Similarly, one can verify that

$$\mathbb{E}\langle \tilde{\boldsymbol{x}}, \boldsymbol{D}\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{D}\tilde{\boldsymbol{x}})\rangle = \sum_{i=1}^{d}\sum_{j=n'+1}^{n} a_j^i \cdot \mathbb{E}[(D_{i,i}\cdot \tilde{x}_i)\cdot \sigma(\langle(\boldsymbol{B}\boldsymbol{D})_{j,:}, \tilde{\boldsymbol{x}}\rangle)]. \tag{C.11}$$

Combining (C.10) and (C.11), and recalling the population risk form in (C.8), we conclude that

$$\mathcal{R}(\boldsymbol{A}, \boldsymbol{B}) \ge \mathcal{R}(\boldsymbol{A}_{:,n'+1:}, \boldsymbol{B}_{n'+1:,:}),$$

where $\boldsymbol{A}_{:,n'+1:}$ and $\boldsymbol{B}_{n'+1:,:}$ are obtained by removing the zero columns/rows from $\boldsymbol{A}$ and $\boldsymbol{B}$, respectively. This means that considering a matrix $\boldsymbol{B}$ with zero rows is equivalent to looking at a smaller rate $r' < r$. We show in Theorem 5, Proposition 5.4.2 and Theorem 8 that the population risk is monotone in the rate. Thus, having zero rows in $\boldsymbol{B}$ is clearly sub-optimal.

C.2 Proofs of Lower Bound on Loss (Section 5.4.1)

C.2.1 Case $r \le 1$

Lower bound on $\widetilde{R}(r)$

Lemma C.2.1. *Let $\boldsymbol{A} = [\boldsymbol{a}_1, \dots, \boldsymbol{a}_n] \in \mathbb{R}^{d\times n}$ and $\boldsymbol{B}^\top = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_n] \in \mathbb{R}^{d\times n}$, with $\|\boldsymbol{b}_i\|_2 = 1$ for $i \in [n]$. Let c_1 and $f(\cdot)$ be defined as per Lemma 5.4.1. Then, the following bound holds:*

$$\mathcal{L}_l(\boldsymbol{A}, \boldsymbol{B}) := \operatorname{Tr}\left[\boldsymbol{A}^\top\boldsymbol{A}\cdot(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)}\right] - \frac{2c_1}{f(1)}\cdot \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right] \ge -\frac{c_1^2}{(f(1))^2}\cdot n. \tag{C.12}$$

Proof of Lemma C.2.1. For any symmetric $\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{T} \in \mathbb{R}^{n\times n}$, a direct computation readily gives that

$$\operatorname{Tr}[\boldsymbol{P}\cdot(\boldsymbol{Q}\circ\boldsymbol{T})] = \operatorname{Tr}[(\boldsymbol{P}\circ\boldsymbol{Q})\cdot\boldsymbol{T}]. \tag{C.13}$$

Thus, by taking $\boldsymbol{P} = \boldsymbol{A}^\top\boldsymbol{A}$, $\boldsymbol{Q} = (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ\ell}$ and $\boldsymbol{T} = (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(\ell+1)}$, we obtain

$$\operatorname{Tr}\left[\boldsymbol{A}^\top\boldsymbol{A}\cdot(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)}\right] = \operatorname{Tr}\left[(\boldsymbol{A}^\top\boldsymbol{A}\circ(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ\ell})\cdot(\boldsymbol{B}\boldsymbol{B}^\top\circ(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ\ell})\right].$$

Note that $\boldsymbol{B}\boldsymbol{B}^\top$ is PSD and, therefore, $(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ\ell}$ is also PSD by Schur product theorem. Furthermore, as the rows of B have unit norm, $(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ\ell}$ has unit diagonal. As a result, if we show that, for any PSD matrix $\boldsymbol{Q}$ with unit diagonal entries,

$$\operatorname{Tr}\left[(\boldsymbol{A}^\top\boldsymbol{A}\circ\boldsymbol{Q})\cdot(\boldsymbol{B}\boldsymbol{B}^\top\circ\boldsymbol{Q})\right] - \frac{2c_1}{f(1)}\cdot\operatorname{Tr}[\boldsymbol{B}\boldsymbol{A}] \geq -\frac{c_1^2}{(f(1))^2}\cdot n, \tag{C.14}$$

then the claim (C.12) immediately follows.

As $\boldsymbol{Q}$ is a PSD matrix with unit diagonal, it admits the following decomposition

$$\boldsymbol{Q} = \sum_{i=1}^n \boldsymbol{u}_i\boldsymbol{u}_i^\top, \quad \boldsymbol{D}_i = \operatorname{Diag}(\boldsymbol{u}_i), \quad \sum_{i=1}^n \boldsymbol{D}_i^2 = \boldsymbol{I}. \tag{C.15}$$

In this view, defining

$$\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{D}_i, \quad \boldsymbol{B}_i = \boldsymbol{D}_i\boldsymbol{B},$$

we can rewrite the LHS of (C.14) in a more convenient form for further analysis. In particular, for the second term we deduce the following

$$\begin{aligned}\operatorname{Tr}[\boldsymbol{B}\boldsymbol{A}] = \operatorname{Tr}[\boldsymbol{A}\boldsymbol{B}] &= \operatorname{Tr}\left[\boldsymbol{A}\cdot\left(\sum_{i=1}^n\boldsymbol{D}_i^2\right)\cdot\boldsymbol{B}\right] = \sum_{i=1}^n\operatorname{Tr}\left[\boldsymbol{A}\cdot\boldsymbol{D}_i^2\cdot\boldsymbol{B}\right]\\ &= \sum_{i=1}^n\operatorname{Tr}[(\boldsymbol{A}\boldsymbol{D}_i)\cdot(\boldsymbol{D}_i\boldsymbol{B})] = \sum_{i=1}^n\operatorname{Tr}[\boldsymbol{A}_i\boldsymbol{B}_i].\end{aligned}$$

Let us now rearrange the first term of (C.14). Notice that

$$\begin{aligned}(\boldsymbol{A}^\top\boldsymbol{A}\circ\boldsymbol{Q})_{i,j} &= \sum_{k=1}^n\langle\boldsymbol{a}_i,\boldsymbol{a}_j\rangle\cdot u_k^iu_k^j = \sum_{k=1}^n\langle\boldsymbol{a}_i\cdot u_k^i,\boldsymbol{a}_j\cdot u_k^j\rangle\\ &= \sum_{k=1}^n((\boldsymbol{A}\boldsymbol{D}_k)^\top\cdot(\boldsymbol{A}\boldsymbol{D}_k))_{i,j} = \sum_{k=1}^n(\boldsymbol{A}_k^\top\boldsymbol{A}_k)_{i,j}.\end{aligned}$$

In the same fashion we get

$$(\boldsymbol{B}\boldsymbol{B}^\top\circ\boldsymbol{Q})_{i,j} = \sum_{k=1}^n(\boldsymbol{B}_k\boldsymbol{B}_k^\top)_{i,j},$$

from which we deduce that

$$\operatorname{Tr}\left[(\boldsymbol{A}^\top\boldsymbol{A}\circ\boldsymbol{Q})\cdot(\boldsymbol{B}\boldsymbol{B}^\top\circ\boldsymbol{Q})\right] = \sum_{i,j=1}^n\operatorname{Tr}\left[\boldsymbol{A}_i^\top\boldsymbol{A}_i\boldsymbol{B}_j\boldsymbol{B}_j^\top\right].$$

Therefore, the proof of (C.14) can be obtained by proving that, for *any* matrices $\boldsymbol{A}_1,\dots,\boldsymbol{A}_n\in\mathbb{R}^{d\times n}$ and $\boldsymbol{B}_1,\dots,\boldsymbol{B}_n\in\mathbb{R}^{n\times d}$,

$$\sum_{i,j=1}^n\operatorname{Tr}\left[\boldsymbol{A}_i^\top\boldsymbol{A}_i\boldsymbol{B}_j\boldsymbol{B}_j^\top\right] - \frac{2c_1}{f(1)}\cdot\sum_{i=1}^n\operatorname{Tr}[\boldsymbol{A}_i\boldsymbol{B}_i] + \frac{c_1^2}{(f(1))^2}\operatorname{Tr}[\boldsymbol{I}] \geq 0. \tag{C.16}$$

To show the last claim, let us define the following matrices

$$\boldsymbol{X} = \sum_{i=1}^{n} \boldsymbol{A}_i^\top \boldsymbol{A}_i, \quad \boldsymbol{Y} = \sum_{i=1}^{n} \boldsymbol{B}_i \boldsymbol{B}_i^\top, \quad \boldsymbol{Z} = \sum_{i=1}^{n} \boldsymbol{B}_i \boldsymbol{A}_i,$$

which allows us to rewrite the statement of (C.16) as

$$\operatorname{Tr}\left[\boldsymbol{X}\boldsymbol{Y} - \frac{2c_1}{f(1)} \cdot \boldsymbol{Z} + \frac{c_1^2}{(f(1))^2} \cdot \boldsymbol{I}\right] \geq 0. \tag{C.17}$$

Note that $\boldsymbol{X}$ is PSD, hence it has a symmetric square root, which we denote by $\sqrt{\boldsymbol{X}}$. Using the continuity of the quantities involved in the LHS of (C.17), we can assume without loss of generality that $\boldsymbol{X}$ is invertible. In fact, the following quantities are continuous: trace, matrix product, matrix transpose. In addition, we can always introduce a small perturbation to $\boldsymbol{A}_i$'s which makes $\boldsymbol{X}$ full-rank. Thus, it suffices to show that (C.17) holds for $\boldsymbol{A}_i$'s such that $\boldsymbol{X}$ is invertible.

In this view, for any matrix $\boldsymbol{T} \in \mathbb{R}^{n \times n}$, we have

$$\begin{aligned}
0 &\leq \sum_{i=1}^{n} \left\| \frac{c_1}{f(1)} \cdot \boldsymbol{T}\boldsymbol{A}_i^\top - \sqrt{\boldsymbol{X}}\boldsymbol{B}_i \right\|_F^2 \\
&= \sum_{i=1}^{n} \operatorname{Tr}\left[\left(\frac{c_1}{f(1)} \cdot \boldsymbol{T}\boldsymbol{A}_i^\top - \sqrt{\boldsymbol{X}}\boldsymbol{B}_i\right) \cdot \left(\frac{c_1}{f(1)} \cdot \boldsymbol{A}_i\boldsymbol{T}^\top - \boldsymbol{B}_i^\top\sqrt{\boldsymbol{X}}\right)\right] \\
&= \sum_{i=1}^{n} \operatorname{Tr}\left[\frac{c_1^2}{(f(1))^2} \cdot \boldsymbol{T}\boldsymbol{A}_i^\top\boldsymbol{A}_i\boldsymbol{T}^\top - \frac{2c_1}{f(1)}\sqrt{\boldsymbol{X}}\boldsymbol{B}_i\boldsymbol{A}_i\boldsymbol{T}^\top + \boldsymbol{X}\boldsymbol{B}_i\boldsymbol{B}_i^\top\right] \\
&= \operatorname{Tr}\left[\frac{c_1^2}{(f(1))^2} \cdot \boldsymbol{T}\boldsymbol{X}\boldsymbol{T}^\top - \frac{2c_1}{f(1)}\sqrt{\boldsymbol{X}}\boldsymbol{Z}\boldsymbol{T}^\top + \boldsymbol{X}\boldsymbol{Y}\right],
\end{aligned} \tag{C.18}$$

where in the second line we used that $\operatorname{Tr}[\boldsymbol{M}] = \operatorname{Tr}\left[\boldsymbol{M}^\top\right]$ for any $\boldsymbol{M}$, and $\operatorname{Tr}[\boldsymbol{M}\boldsymbol{N}] = \operatorname{Tr}[\boldsymbol{N}\boldsymbol{M}]$ for any $\boldsymbol{M}, \boldsymbol{N}$.

As $\boldsymbol{X}$ is invertible, its square root $\sqrt{\boldsymbol{X}}$ is invertible. As $\boldsymbol{X}$ is also PSD, its inverse, i.e., $\boldsymbol{X}^{-1}$, is PSD and, hence, it has a symmetric square root, i.e., $\sqrt{\boldsymbol{X}^{-1}}$. In this view, we get that

$$\sqrt{\boldsymbol{X}^{-1}} = (\sqrt{\boldsymbol{X}})^{-1}.$$

Thus, by picking $\boldsymbol{T} = (\sqrt{\boldsymbol{X}})^{-1}$, we obtain

$$\boldsymbol{T}^\top\boldsymbol{T} = \boldsymbol{T}^2 = \boldsymbol{X}^{-1}, \quad \boldsymbol{T}^\top\sqrt{\boldsymbol{X}} = \boldsymbol{T}\sqrt{\boldsymbol{X}} = \boldsymbol{I}.$$

Using these observations, we deduce that the RHS of (C.18) is equal to the LHS of (C.17), which concludes the proof. □

Matrices in $\mathcal{H}_{n,d}$ Are the Only Minimizers

Lemma C.2.2. *Let $\boldsymbol{A} \in \mathbb{R}^{d \times n}$ and $\boldsymbol{B}^\top = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n] \in \mathbb{R}^{d \times n}$, with $\|\boldsymbol{b}_i\|_2 = 1$ for $i \in [n]$. Let c_1 and $f(\cdot)$ be defined as per Lemma 5.4.1. Then, we have that the set of minimizers of*

$$\operatorname{Tr}\left[\boldsymbol{A}^\top\boldsymbol{A} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2c_1 \cdot \operatorname{Tr}[\boldsymbol{B}\boldsymbol{A}] \tag{C.19}$$

coincides with the set $\mathcal{H}_{n,d}$ of weight-tied orthogonal matrices .

Proof of Lemma C.2.2. A direct computation immediately shows that the lower bound (C.12) is achieved for all $\ell \in \mathbb{N}$ by matrices $(\boldsymbol{A}, \boldsymbol{B})$ that belong to the set $\mathcal{H}_{d,n}$. Define the sets of minimizers of (C.12) as follows

$$\begin{aligned}\mathcal{M}_\ell :&= \underset{\boldsymbol{A},\boldsymbol{B}:\|\boldsymbol{b}_i\|_2=1}{\arg\min}\ \mathcal{L}_\ell(\boldsymbol{A}, \boldsymbol{B}) \\ &= \left\{(\boldsymbol{A}_{\boldsymbol{B}}, \boldsymbol{B}) : \boldsymbol{A}_{\boldsymbol{B}} \in \underset{\boldsymbol{A}}{\arg\min}\, \mathcal{L}_\ell(\boldsymbol{A}, \boldsymbol{B}),\ \boldsymbol{B} \in \underset{\boldsymbol{B}:\|\boldsymbol{b}_i\|_2=1}{\arg\min}\, \mathcal{L}_\ell(\boldsymbol{A}_{\boldsymbol{B}}, \boldsymbol{B})\right\}.\end{aligned}$$

We will now show that

$$\bigcap_{l=0}^{\infty} \mathcal{M}_\ell = \mathcal{H}_{n,d}. \tag{C.20}$$

As the Taylor coefficients of $f(\cdot)$ are non-negative, (C.20) readily gives that the set of minimizers of (C.19) coincides with $\mathcal{H}_{n,d}$. Futher, recall that $c_1 \neq 0$ and $\sum_{\ell=1}^{\infty}(c_{2\ell+1})^2 \neq 0$ and, hence, (C.20) is the union of the linear term ($\ell = 0$) and at least one non-linear ($\ell > 0$) term.

We first prove that it is enough to consider the case $r = 1$. Thus, assume that the result holds for $n = d$ and consider now $n < d$. We have that, for any orthogonal matrix $\boldsymbol{O} \in \mathbb{R}^{d\times d}$,

$$\begin{aligned}\mathbb{E}_{\boldsymbol{x}} \|\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{x})\|_2^2 &= \mathbb{E}_{\boldsymbol{x}} \|\boldsymbol{O}\boldsymbol{x} - \boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{O}\boldsymbol{x})\|_2^2 \\ &= \mathbb{E}_{\boldsymbol{x}} \left\|\boldsymbol{x} - \boldsymbol{O}^\top\boldsymbol{A}\sigma(\boldsymbol{B}\boldsymbol{O}\boldsymbol{x})\right\|_2^2,\end{aligned} \tag{C.21}$$

where in the first step we have used the rotational invariance of $\boldsymbol{x}$, and in the second step we have multiplied the argument of the norm by the orthogonal matrix $\boldsymbol{O}^\top$. Thus, (C.21) gives that $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{H}_{n,d}$ if and only if $(\boldsymbol{O}^\top\boldsymbol{A}, \boldsymbol{B}\boldsymbol{O}) \in \mathcal{H}_{n,d}$.

Let us write the SVD of $\boldsymbol{B}$ as $\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$, where $\boldsymbol{U} \in \mathbb{R}^{n\times n}, \boldsymbol{V} \in \mathbb{R}^{d\times d}$ are orthogonal matrices and $\boldsymbol{D} \in \mathbb{R}^{n\times d}$ is a (rectangular) diagonal matrix. Thus, by taking $\boldsymbol{O} = \boldsymbol{V}$, one can assume that $\boldsymbol{B}$ has the form $(\boldsymbol{B}_{1:n,1:n}, \boldsymbol{0}_{1:n,1:d-n})$, where $\boldsymbol{B}_{1:n,1:n}$ denotes the left $n \times n$ sub-matrix of $\boldsymbol{B}$ and $\boldsymbol{0}_{1:n,1:d-n}$ denotes a $n \times (d - n)$ matrix of 0's. We also write the decompositions $\boldsymbol{A} = ((\boldsymbol{A}_{1:n,1:n})^\top, (\boldsymbol{A}_{n+1:d,1:n})^\top)^\top$ and $\boldsymbol{x} = (\boldsymbol{x}_{1:n}, \boldsymbol{x}_{n+1:d})$, where $\boldsymbol{A}_{1:n,1:n}$ (resp. $\boldsymbol{A}_{n+1:d,1:n}$) denotes the top $n \times n$ (resp. bottom $(d - n) \times n$) sub-matrix of $\boldsymbol{A}$, and $\boldsymbol{x}_{1:n}$ (resp. $\boldsymbol{x}_{n+1:d}$) denotes the first n (resp. last $d - n$) components of $\boldsymbol{x}$. Hence, the objective (5.2) can be expressed (up to the constant multiplicative factor d^{-1}) as the sum of

$$\mathcal{R}_1(\boldsymbol{A}, \boldsymbol{B}) = \mathbb{E}\left[\|\boldsymbol{x}_{1:n} - \boldsymbol{A}_{1:n,1:n}\sigma(\boldsymbol{B}_{1:n,1:n}\boldsymbol{x}_{1:n})\|^2\right]$$

and

$$\mathcal{R}_2(\boldsymbol{A}, \boldsymbol{B}) = \mathbb{E}\left[\|\boldsymbol{x}_{n+1:d} - \boldsymbol{A}_{n+1:d,1:n}\sigma(\boldsymbol{B}_{1:n,1:n}\boldsymbol{x}_{1:n})\|^2\right].$$

As $\boldsymbol{x}_{n+1:d}$ has zero mean and it is independent from $\boldsymbol{x}_{1:n}$, we have that

$$\mathcal{R}_2(\boldsymbol{A}, \boldsymbol{B}) = d - n + \mathbb{E}\left[\|\boldsymbol{A}_{n+1:d,1:n}\sigma(\boldsymbol{B}_{1:n,1:n}\boldsymbol{x}_{1:n})\|^2\right],$$

which is minimized by setting $\boldsymbol{A}_{n+1:d,1:n}$ to $\boldsymbol{0}$. Note that $\mathcal{R}_1$ depends only on $\boldsymbol{A}_{1:n,1:n}, \boldsymbol{B}_{1:n,1:n}$ (and not on $\boldsymbol{A}_{n+1:d,1:n}$), hence its minimizers are $(\boldsymbol{A}_{1:n,1:n}, \boldsymbol{B}_{1:n,1:n}) \in \mathcal{H}_{n,n}$ by our assumption on the $r = 1$ case. As a result, by using that $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{H}_{d,n}$ if and only if $(\boldsymbol{O}^\top\boldsymbol{A}, \boldsymbol{B}\boldsymbol{O}) \in \mathcal{H}_{d,n}$, we conclude that all the minimizers of the desired objective have the form $\boldsymbol{O}((\boldsymbol{A}_{1:n,1:n})^\top, (\boldsymbol{0}_{1:n-d,1:n})^\top)^\top$ and $(\boldsymbol{B}_{1:n,1:n}, \boldsymbol{0}_{1:n,1:d-n})\boldsymbol{O}^\top$, i.e., they form the set $\mathcal{H}_{n,d}$ defined in (5.7).

It remains to prove the result for $r = 1$. First, consider $\ell = 0$. In this case, we have

$$\begin{aligned}\mathcal{L}_0(\boldsymbol{A}, \boldsymbol{B}) &= \operatorname{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{B}\boldsymbol{B}^\top\right] - \frac{2c_1}{f(1)} \cdot \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right] \\ &= \operatorname{Tr}\left[\boldsymbol{B}^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{B}\right] - \frac{2c_1}{f(1)} \cdot \operatorname{Tr}\left[\boldsymbol{A}\boldsymbol{B}\right] \\ &= \|\boldsymbol{A}\boldsymbol{B}\|_F^2 - \frac{2c_1}{f(1)} \cdot \operatorname{Tr}\left[\boldsymbol{A}\boldsymbol{B}\right], \end{aligned} \tag{C.22}$$

where we have used that the trace is invariant under cyclic permutation. Notice that the minimizer of (C.22) is clearly $\boldsymbol{A}\boldsymbol{B} = \frac{c_1}{f(1)}\boldsymbol{I}_d$.

Consider some $\ell \geq 1$. As $\boldsymbol{A}\boldsymbol{B} = \frac{c_1}{f(1)}\boldsymbol{I}_d$ and $\boldsymbol{A}, \boldsymbol{B}$ are square matrices, $\boldsymbol{B}$ is invertible and $\boldsymbol{A}^\top\boldsymbol{A} = \frac{c_1^2}{(f(1))^2} \cdot (\boldsymbol{B}\boldsymbol{B}^\top)^{-1}$. Thus,

$$\begin{aligned}\mathcal{L}_\ell(\boldsymbol{A}, \boldsymbol{B}) &= \operatorname{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A}(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)}\right] - \frac{2c_1}{f(1)} \cdot \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right] \\ &= \frac{c_1^2}{(f(1))^2} \cdot \operatorname{Tr}\left[(\boldsymbol{B}\boldsymbol{B}^\top)^{-1}(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)}\right] - \frac{2c_1^2}{(f(1))^2} \cdot n. \end{aligned} \tag{C.23}$$

Let $\boldsymbol{P} = \boldsymbol{B}\boldsymbol{B}^\top$. Note that $\boldsymbol{P}$ is symmetric and, hence, also its inverse is symmetric. Then, by using (C.13), we have that

$$\operatorname{Tr}\left[\boldsymbol{P}^{-1}\boldsymbol{P}^{\circ(2\ell+1)}\right] = \operatorname{Tr}\left[(\boldsymbol{P}^{-1} \circ \boldsymbol{P})\boldsymbol{P}^{\circ 2\ell}\right]. \tag{C.24}$$

An application of Theorem 5 in [Vis00] gives that

$$\boldsymbol{P} \circ \boldsymbol{P}^{-1} \succeq \boldsymbol{I}, \tag{C.25}$$

where $\succeq$ denotes majorization in the PSD sense. We now show that $\boldsymbol{P} \circ \boldsymbol{P}^{-1} = \boldsymbol{I}$. To do so, suppose by contradiction that

$$\boldsymbol{P} \circ \boldsymbol{P}^{-1} = \boldsymbol{I} + \boldsymbol{R},$$

for some $\boldsymbol{R} \succeq \boldsymbol{0}$ such that $\boldsymbol{R} \neq \boldsymbol{0}$. Hence,

$$\operatorname{Tr}\left[(\boldsymbol{P}^{-1} \circ \boldsymbol{P})\boldsymbol{P}^{\circ 2\ell}\right] = \operatorname{Tr}\left[\boldsymbol{P}^{\circ 2\ell}\right] + \operatorname{Tr}\left[\boldsymbol{R}\boldsymbol{P}^{\circ 2\ell}\right] = n + \operatorname{Tr}\left[\boldsymbol{R}\boldsymbol{P}^{\circ 2\ell}\right], \tag{C.26}$$

where in the last equality we use that $\boldsymbol{P}$ (and, consequently, $\boldsymbol{P}^{\circ 2\ell}$) has unit diagonal. By the Schur product theorem, $\boldsymbol{P}^{\circ 2\ell} \succ \boldsymbol{0}$ and, hence, it admits a square root. Thus, we get

$$\operatorname{Tr}\left[\boldsymbol{R}\boldsymbol{P}^{\circ 2\ell}\right] = \operatorname{Tr}\left[\sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{R} \cdot \sqrt{\boldsymbol{P}^{\circ 2\ell}}\right].$$

It is easy to see that the matrix $\sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{R} \cdot \sqrt{\boldsymbol{P}^{\circ 2\ell}}$ is PSD and, thus,

$$\operatorname{Tr}\left[\sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{R} \cdot \sqrt{\boldsymbol{P}^{\circ 2\ell}}\right] \geq 0,$$

where the inequality is strict if and only if the corresponding matrix has only zero eigenvalues. However, for any non-zero $\boldsymbol{v} \in \mathbb{R}^n$, we have that

$$\boldsymbol{u_v} := \sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{v} \neq 0,$$

since $\sqrt{\boldsymbol{P}^{\circ 2\ell}}$ is strictly positive definite (as $\boldsymbol{P}^{\circ 2\ell} \succ \boldsymbol{0}$) and, thus, it does not have 0 eigenvalues. Hence, if

$$\boldsymbol{v}^\top \cdot \sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{R} \cdot \sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{v} = \boldsymbol{u_v}^\top \boldsymbol{R}\boldsymbol{u_v} = 0,$$

then $\boldsymbol{u}_v \neq \mathbf{0}$ is an eigenvector of $\boldsymbol{R}$ corresponding to a zero eigenvalue. In this view, if $\sqrt{\boldsymbol{P}^{\circ 2\ell}} \cdot \boldsymbol{R} \cdot \sqrt{\boldsymbol{P}^{\circ 2\ell}}$ has all zero eigenvalues, then all eigenvalues of $\boldsymbol{R}$ are zero. As $\boldsymbol{R}$ cannot be the zero matrix, by using (C.26), we conclude that

$$\mathrm{Tr}\left[(\boldsymbol{P}^{-1} \circ \boldsymbol{P})\boldsymbol{P}^{\circ 2\ell}\right] > n. \tag{C.27}$$

By combining (C.23), (C.24) and (C.27), we have that $\mathcal{L}_\ell(\boldsymbol{A}, \boldsymbol{B}) > -c_1^2 n/(f(1))^2$, which contradicts with the fact that $(\boldsymbol{A}, \boldsymbol{B})$ is a minimizer (since any $(\boldsymbol{A}', \boldsymbol{B}') \in \mathcal{H}_{n,d}$ achieves the value of $-c_1^2 n/(f(1))^2$). Therefore, we conclude that $\boldsymbol{P} \circ \boldsymbol{P}^{-1} = \boldsymbol{I}$.

At this point, we show that $\boldsymbol{P} \circ \boldsymbol{P}^{-1} = \boldsymbol{I}$ implies that $\boldsymbol{P} = \boldsymbol{I}$. Note that $\boldsymbol{P}$ is a Gram matrix, and let its basis be $\{\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n\}$. Define

$$\boldsymbol{b}_i' = \boldsymbol{b}_i - \tilde{\boldsymbol{b}}_i,$$

where $\tilde{\boldsymbol{b}}_i$ is orthogonal projection of $\boldsymbol{b}_i$ onto the space spanned by $\{\boldsymbol{b}_j\}_{j\neq i}^n$. From a well-known result (see, for instance, Theorem 2.1 in [dPG95]) we have that

$$\boldsymbol{P}_{ii}^{-1} = \frac{1}{\|\boldsymbol{b}_i'\|_2^2}. \tag{C.28}$$

Hence, we obtain that

$$\|\boldsymbol{b}_i'\|_2 \leq \|\boldsymbol{b}_i\|_2 = 1, \tag{C.29}$$

where the inequality is sharp only if $\boldsymbol{b}_i$ is orthogonal to all $\{\boldsymbol{b}_j\}_{j\neq i}^n$. Then, from (C.28), we deduce

$$n = \mathrm{Tr}\left[\boldsymbol{I}\right] = \mathrm{Tr}\left[\boldsymbol{P} \circ \boldsymbol{P}^{-1}\right] = \sum_{i=1}^n \|\boldsymbol{b}_i\|_2^2 \cdot \frac{1}{\|\boldsymbol{b}_i'\|_2^2} = \sum_{i=1}^n \frac{1}{\|\boldsymbol{b}_i'\|_2^2}. \tag{C.30}$$

By combining (C.29) and (C.30), we conclude that $\{\boldsymbol{b}_i\}_{i\in[n]}$ form an orthonormal basis, and, hence, $\boldsymbol{P} = \boldsymbol{I}$. This means that (C.20) holds for $r = 1$ since

$$\text{(C.19)} = \sum_{\ell=1}^\infty (c_{2\ell+1})^2 \cdot \mathcal{L}_\ell(\boldsymbol{A}, \boldsymbol{B}),$$

which concludes the proof. □

Proof of Theorem 5. It follows by combining the results of Lemma C.2.1 and C.2.2. □

C.2.2 Case $r > 1$

Lower bound on $\widetilde{R}(r)$

Proof of Proposition 5.4.2. An application of Theorem A in [Kha21] gives that

$$\begin{aligned}\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{B}\boldsymbol{B}^\top\right] &= \langle \mathbf{1}, (\boldsymbol{A}^\top \boldsymbol{A} \circ \boldsymbol{B}\boldsymbol{B}^\top)\mathbf{1}\rangle \\ &\geq \frac{1}{d}\langle \mathbf{1}, (\mathrm{Diag}(\boldsymbol{B}\boldsymbol{A})\mathrm{Diag}(\boldsymbol{B}\boldsymbol{A})^\top)\mathbf{1}\rangle = \frac{1}{d}\left(\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right]\right)^2,\end{aligned}$$

where $\mathrm{Diag}(\boldsymbol{B}\boldsymbol{A}) \in \mathbb{R}^n$ stands for the vector with entries corresponding to the diagonal of the matrix $\boldsymbol{B}\boldsymbol{A}$. Hence, we have

$$\begin{aligned}&\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right] \\ &\quad \geq \frac{c_1^2}{d}\left(\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right]\right)^2 + \sum_{\ell=1}^\infty (c_{2\ell+1})^2 \cdot \mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ 2\ell+1}\right] - 2c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right].\end{aligned} \tag{C.31}$$

Define $\alpha := f(1) - c_1^2$. Then, for any $\beta \in [0, 1]$, we can rewrite the RHS of (C.31) as

$$\begin{aligned} &\left[\frac{c_1^2}{d} \left(\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A} \right] \right)^2 - 2(1-\beta)c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A} \right] \right] \\ &\qquad + \sum_{\ell=1}^{\infty} (c_{2\ell+1})^2 \cdot \left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ 2\ell+1} \right] - \frac{2\beta c_1}{\alpha} \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A} \right] \right). \end{aligned} \tag{C.32}$$

The first term in (C.32) is a quadratic polynomial in $\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A}\right]$. Hence, we have that

$$\left[\frac{c_1^2}{d} \left(\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A} \right] \right)^2 - 2(1-\beta)c_1 \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A} \right] \right] \geq -d(1-\beta)^2. \tag{C.33}$$

Define $\boldsymbol{B}_e := [\boldsymbol{B}, \boldsymbol{0}_{1:n,1:n-d}]$ and $\boldsymbol{A}_e^\top := [\boldsymbol{A}^\top, \boldsymbol{0}_{1:n,1:n-d}]$. One can readily verify that the traces in the second term of (C.32) remain unchanged if we replace $\boldsymbol{A}$ and $\boldsymbol{B}$ with $\boldsymbol{A}_e$ and $\boldsymbol{B}_e$, respectively. Note that $\boldsymbol{A}_e, \boldsymbol{B}_e$ are square matrices, hence we can apply Lemma C.2.1 (which readily generalizes to a different scaling in front of the second trace) to get

$$\begin{aligned} &\sum_{\ell=1}^{\infty} (c_{2\ell+1})^2 \cdot \left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ 2\ell+1} \right] - \frac{2\beta c_1}{\alpha} \cdot \mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{A} \right] \right) \\ &\qquad \geq - \sum_{\ell=1}^{\infty} (c_{2\ell+1})^2 \cdot \frac{\beta^2 c_1^2}{\alpha^2} n = - \frac{\beta^2 c_1^2}{\alpha} n. \end{aligned} \tag{C.34}$$

By combining (C.31), (C.32), (C.33) and (C.34), we obtain that

$$\frac{1}{d} \left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top) \right] - 2 \cdot \mathrm{Tr}\left[\boldsymbol{A}\boldsymbol{B} \right] \right) + 1 \geq 1 - (1-\beta)^2 - \frac{\beta^2 c_1^2}{\alpha} r. \tag{C.35}$$

By taking $\beta = \alpha/(c_1^2 r + \alpha)$ and re-arranging the RHS of (C.35), the desired result readily follows. □

Asymptotic Achievability of the Lower Bound

Lemma C.2.3. *Let $\boldsymbol{A}, \boldsymbol{B}$ be defined as in* (5.12)*. Then, for any $\epsilon > 0$, we have that, with probability at least $1 - c/d^2$,*

$$\left| \left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} f(\boldsymbol{B}\boldsymbol{B}^\top) \right] - 2c_1 \mathrm{Tr}\left[\boldsymbol{A}\boldsymbol{B} \right] \right) - \left(\beta^2 c_1^2 r n + \beta^2 \alpha n - 2c_1 \beta n \right) \right| \leq C n^{\frac{1}{2}+\epsilon}.$$

Thus, choosing $\beta = \frac{c_1}{c_1^2 r + \alpha}$ the loss approaches $1 - \frac{r}{r + \frac{\alpha}{c_1^2}}$, i.e., with the same probability,

$$\left| \left(1 + \frac{1}{d} \left(\mathrm{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} f(\boldsymbol{B}\boldsymbol{B}^\top) \right] - 2c_1 \mathrm{Tr}\left[\boldsymbol{A}\boldsymbol{B} \right] \right) \right) - \left(1 - \frac{r}{r + \frac{\alpha}{c_1^2}} \right) \right| \leq C d^{-\frac{1}{2}+\epsilon}.$$

Here, the constants c, C depend only on r and ϵ.

We start by proving the following.

Lemma C.2.4. *Let $\hat{\boldsymbol{B}}, \boldsymbol{B}$ be defined as in* (5.12)*. Then, for any $\epsilon > 0$, we have that, with probability at least $1 - c/d^2$,*

$$\max_{i,j} \left| \frac{(\boldsymbol{B}\boldsymbol{B}^\top)_{i,j}}{(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)_{i,j}} - 1 \right| \leq C n^{-\frac{1}{2}+\epsilon}.$$

Here, the constants c, C depend only on r and ϵ.

Proof. If $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ is sampled uniformly from $\mathbb{SO}(n)$, then it follows from rotational invariance that any fixed row or column is uniformly distributed on the n-dimensional sphere $\mathbb{S}^{n-1}$. Thus, any fixed row of $\boldsymbol{U}$ is distributed as $\boldsymbol{g}/\left\|\boldsymbol{g}\right\|_2$, where $\boldsymbol{g} \sim \mathcal{N}\left(0, \boldsymbol{I}/n\right)$. Now, it follows from the concentration of $\left\|\boldsymbol{g}\right\|_2$ (see e.g. Theorem 3.1.1 in [Ver18]) that $\left\|\left\|\boldsymbol{g}\right\|_2 - 1\right\|_{\psi_2} \le Cn^{-\frac{1}{2}}$, where $\left\|\cdot\right\|_{\psi_2}$ denotes the sub-Gaussian norm. Denote by $\boldsymbol{g}_d \in \mathbb{R}^d$ the first d components of $\boldsymbol{g}_d$. Then, by the same reasoning, it holds that $\left\|\sqrt{r}\left\|\boldsymbol{g}_d\right\|_2 - 1\right\|_{\psi_2} \le cd^{-\frac{1}{2}}$. Looking at the definition of $\hat{\boldsymbol{B}}$, we have that, for any fixed i, the distribution of its rows is given by $\hat{\boldsymbol{b}}_i \sim \sqrt{r}\boldsymbol{g}_d/\left\|\boldsymbol{g}\right\|_2$. Furthermore, for any pair of indices i, j, we have that

$$\frac{(\boldsymbol{B}\boldsymbol{B}^\top)_{i,j}}{(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)_{i,j}} = \frac{1}{\|\hat{\boldsymbol{b}}_i\|_2 \cdot \|\hat{\boldsymbol{b}}_j\|_2}.$$

Hence,

$$\mathbb{P}\left(\left|\frac{(\boldsymbol{B}\boldsymbol{B}^\top)_{i,j}}{(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)_{i,j}} - 1\right| \le n^{-\frac{1}{2}+\epsilon}\right) = \mathbb{P}\left(\left|\frac{1}{\|\hat{\boldsymbol{b}}_i\|_2 \cdot \|\hat{\boldsymbol{b}}_j\|_2} - 1\right| \le n^{-\frac{1}{2}+\epsilon}\right) \le C\exp\left(-\frac{d^\epsilon}{C}\right).$$

Now a simple union bound over all rows gives us

$$\mathbb{P}\left(\max_{i,j}\left|\frac{1}{\|\hat{\boldsymbol{b}}_i\|_2 \cdot \|\hat{\boldsymbol{b}}_j\|_2} - 1\right| \le n^{-\frac{1}{2}+\epsilon}\right) \le Cn\exp\left(-\frac{d^\epsilon}{C}\right) \le \frac{C}{d^2},$$

which implies the desired result. □

Next, we bound the traces of the terms $\boldsymbol{B}\boldsymbol{B}^\top(\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)}$. We start with the case $\ell = 0$.

Lemma C.2.5. *Let $\boldsymbol{B}$ be defined as in* (5.12)*. Then, for any $\epsilon > 0$, with probability at least $1 - c/d^2$,*

$$\left|\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top(\boldsymbol{B}\boldsymbol{B}^\top)\right] - rn\right| \le Cd^{\frac{1}{2}+\epsilon}.$$

Here, the constants c, C depend only on r and ϵ.

Proof. Note that

$$\begin{aligned}\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top(\boldsymbol{B}\boldsymbol{B}^\top)\right] &= \sum_{i,j}\left((\boldsymbol{B}\boldsymbol{B}^\top)_{i,j}\right)^2 \\ &= \sum_{i,j}\left(\frac{\left((\boldsymbol{B}\boldsymbol{B}^\top)_{i,j}\right)^2}{\left((\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)_{i,j}\right)^2} - 1\right)\left((\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)_{i,j}\right)^2 + \mathrm{Tr}\left[\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)\right].\end{aligned}$$

Thus, an application of Lemma C.2.4 gives that, with probability at least $1 - c/d^2$,

$$\left|\mathrm{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top(\boldsymbol{B}\boldsymbol{B}^\top)\right] - \mathrm{Tr}\left[\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)\right]\right| \le \mathrm{Tr}\left[\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)\right] \cdot Cd^{-\frac{1}{2}+\epsilon}. \tag{C.36}$$

Since the trace is invariant under cyclic permutation, we readily have that

$$\mathrm{Tr}\left[\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top(\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)\right] = rn. \tag{C.37}$$

By combining (C.36) and (C.37), the desired result follows. □

Finally, we consider the higher order terms for $\ell \geq 1$.

Lemma C.2.6. *Let $\boldsymbol{B}$ be defined as in* (5.12)*. Then, for any $\epsilon > 0$, we have that, with probability at least $1 - c/d^2$,*

$$\sup_{\ell \geq 1} \left| \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)} \right] - n \right| \leq C \log^2 n.$$

Here, the constants c, C depend only on r and ϵ.

Proof. We first observe that

$$\operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)} \right] = \sum_{i,j} \left((\boldsymbol{B}\boldsymbol{B}^\top)_{i,j} \right)^{2\ell+2} = n + \sum_{i \neq j} \left((\boldsymbol{B}\boldsymbol{B}^\top)_{i,j} \right)^{2\ell+2}.$$

An application of Lemma C.2.4 gives that, with probability $1 - c/d^2$,

$$\sup_{\ell \geq 1} \sum_{i \neq j} \left((\boldsymbol{B}\boldsymbol{B}^\top)_{i,j} \right)^{2\ell+2} \leq \sup_{\ell \geq 1} \sum_{i \neq j} \left((1 + Cd^{-1/2+\epsilon}) \cdot (\hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top)_{i,j} \right)^{2\ell+2}. \tag{C.38}$$

Furthermore, by using the first part of Lemma C.5.2 with $\boldsymbol{A} = \hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top$, we have that, with probability at least $1 - 1/n^2$, the RHS of (C.38) is lower bounded by

$$\sup_{\ell \geq 1} \sum_{i \neq j} \left((1 + Cd^{-1/2+\epsilon}) \cdot C\sqrt{\frac{\log n}{n}} \right)^{2\ell+2} \leq C \log^2 n,$$

which implies the desired result. □

At this point, we are ready to give the proof of Lemma C.2.3.

Proof of Lemma C.2.3. Recall that $\{(c_{2\ell+1})^2\}_{\ell=0}^\infty$ denote the Taylor coefficients of $f(x)$. By using that $\boldsymbol{A} = \beta \boldsymbol{B}^\top$, our objective becomes

$$\begin{aligned}
\operatorname{Tr}\left[\boldsymbol{A}^\top \boldsymbol{A} f(\boldsymbol{B}\boldsymbol{B}^\top) \right] - 2c_1 \operatorname{Tr}\left[\boldsymbol{A}\boldsymbol{B} \right] &= \beta^2 \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top f(\boldsymbol{B}\boldsymbol{B}^\top) \right] - 2c_1 \beta n \\
&= \beta^2 \sum_{\ell=0}^\infty (c_{2\ell+1})^2 \operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)} \right] - 2c_1 \beta n \\
&= \beta^2 c_1^2 rn + \beta^2 \sum_{\ell=1}^\infty (c_{2\ell+1})^2 n - 2c_1 \beta n \\
&\quad + \beta^2 c_1^2 \left(\operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top (\boldsymbol{B}\boldsymbol{B}^\top) \right] - rn \right) + \beta^2 \sum_{\ell=1}^\infty (c_{2\ell+1})^2 \left(\operatorname{Tr}\left[\boldsymbol{B}\boldsymbol{B}^\top (\boldsymbol{B}\boldsymbol{B}^\top)^{\circ(2\ell+1)} \right] - n \right).
\end{aligned}$$

Then, by bounding the last two terms with Lemma C.2.5 and Lemma C.2.6, the desired result follows. □

Proof of Proposition 5.4.3. The proof is a direct application of Lemma C.2.3. □

C.3 Global Convergence of Weight-tied Gradient Flow (Theorem 6)

We start by giving a recap of the weight-tied gradient flow considered in Section 5.4.2. Under the weight-tying constraint (5.14), the objective (5.13) has the following form

$$\begin{aligned}\Psi(\beta, \boldsymbol{B}) &:= \beta^2 \cdot \mathrm{Tr}\left[\boldsymbol{B}^\top \boldsymbol{B} \cdot f(\boldsymbol{B}\boldsymbol{B}^\top)\right] - 2\beta n \\ &= \beta^2 \cdot \sum_{i,j=1}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right) - 2\beta n,\end{aligned} \tag{C.39}$$

where $\|\boldsymbol{b}_i\|_2 = 1$ for all i. Note that the optimal β^* can be found exactly, since (C.39) is a quadratic polynomial in β. In this view, to optimize (C.39), we perform a gradient flow on $\{\boldsymbol{b}_i\}_{i=1}^n$, which are regarded as vectors on the unit sphere, and pick the optimal β^* at each time t. Formally,

$$\begin{aligned}\beta(t) &= \frac{n}{\sum_{i,j=1}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right)}, \\ \frac{\partial \boldsymbol{b}_i(t)}{\partial t} &= -\boldsymbol{J}_i(t) \nabla_{\boldsymbol{b}_i} \Psi(\beta(t), \boldsymbol{B}(t)),\end{aligned} \tag{C.40}$$

where $\boldsymbol{J}_i(t) := \boldsymbol{I} - \boldsymbol{b}_i(t)\boldsymbol{b}_i(t)^\top$ projects the gradient $\nabla_{\boldsymbol{b}_i} \Psi(\beta(t), \boldsymbol{B}(t))$ onto the tangent space at the point $\boldsymbol{b}_i(t)$ (see (C.45) for the closed form expression). This ensures that $\|\boldsymbol{b}_i(t)\|_2 = 1$ along the gradient flow trajectory. The described procedure can be viewed as Riemannian gradient flow, due to the projection of the gradient $\nabla_{\boldsymbol{b}_i} \Psi(\beta(t), \boldsymbol{B}(t))$ on the tangent space of the unit sphere. We now recap the statement of Theorem 6.

Theorem 11. *Fix $r \leq 1$. Let $\boldsymbol{B}(t)$ be obtained via the gradient flow* (C.40) *applied to Ψ defined in* (C.39). *Let the initialization $\boldsymbol{B}(0)$ have unit-norm rows and $\mathrm{rank}(\boldsymbol{B}(0)) = n$. Then, as $t \to \infty$, $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ converges to $\boldsymbol{I}$, which is the unique global optimum of* (C.39). *Moreover, define the residual*

$$\phi(t) = \mathrm{Tr}\left[(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top - \boldsymbol{I}) \cdot f(\boldsymbol{B}(t)^\top \boldsymbol{B}(t))\right] \geq 0,$$

which vanishes at the minimizer, and let T be the first time such that $\phi(T) = \delta$. Then,

$$T \leq -\mathbb{1}\{\phi(0) > nf(1)\} \cdot f(1) \cdot \log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) \tag{C.41}$$

$$-\mathbb{1}\{\delta \leq nf(1)\} \cdot \frac{2f^2(1)}{\delta} \cdot \log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top). \tag{C.42}$$

We now are ready to present the proof of Theorem 11. Let $\boldsymbol{B}^\top = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n]$. Recall that, under the weight-tying (5.14), the objective in (5.13) can be re-written as

$$\beta^2 \cdot \sum_{i,j=1}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right) - 2\beta n. \tag{C.43}$$

By the definition in Theorem 11, the residual $\phi(t)$ is given by

$$\phi(t) := \sum_{i \neq j}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right). \tag{C.44}$$

In this view, in accordance with (C.40), we study the following gradient flow:

$$\begin{cases} \frac{\partial \boldsymbol{b}_k(t)}{\partial t} = -\beta^2(t) \cdot \left[\boldsymbol{J}_k(t) \sum_{i \neq j} \boldsymbol{b}_j(t) \cdot g(\langle \boldsymbol{b}_k(t), \boldsymbol{b}_j(t) \rangle) \right], \\ \beta(t) = \dfrac{n}{nf(1) + \phi(t)}, \\ \|\boldsymbol{b}_k(0)\|_2 = 1, \end{cases} \tag{C.45}$$

where $g(x) := x \cdot f'(x) + f(x)$, and we have rescaled the time of the dynamics by a factor 2 to omit the factor 2 in front of $\beta^2(t)$. From here on, we will suppress the time notation when it is clear from the context, for the sake of simplicity. Note that one of the terms is absent in the summation, due to the fact that by definition of the operator $\boldsymbol{J}_k$:

$$\boldsymbol{J}_k \boldsymbol{b}_k = \boldsymbol{0}.$$

In addition, since $\boldsymbol{J}_k$ defines the projection of the gradient on the tangent space at the point $\boldsymbol{b}_k$ of the unit sphere, along the trajectory of the gradient flow (C.45) we have that $\|\boldsymbol{b}_k\|_2 = 1$.

The gradient flow (C.45) is well-defined (i.e., its solution exists and it is unique) when its RHS is Lipschitz continuous (see, for instance, [San17]). It suffices to check the Lipschitz continuity of $g(\cdot)$. Note that both $xf'(x)$ and $f(x)$ are Lipschitz continuous on any interval $[-1+\delta, 1-\delta]$ for some $\delta > 0$. Hence, the RHS of (C.45) is Lipschitz continuous, if

$$\max_{i \neq j} |\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle| \leq 1 - \delta, \tag{C.46}$$

where δ is bounded away from 0 uniformly in t.

Recall that, by the assumption of Theorem 11, we have that $\mathrm{rank}(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) = n$, hence $\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) \geq \varepsilon_1$ for some $\varepsilon_1 > 0$. Thus, from the result in Lemma C.3.2, we obtain that

$$\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) \geq \varepsilon_1. \tag{C.47}$$

Let $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n$ denote the eigenvalues of $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ in increasing order. Then, (C.47) directly gives that

$$\lambda_1 \prod_{i=2}^{n} \lambda_i \geq \varepsilon_1 > 0.$$

Since $\boldsymbol{B}(t)\boldsymbol{B}(t)^\top$ has unit diagonal, we have that $\sum_{i=1}^n \lambda_i = n$. Hence, the smallest possible value of λ_1 during the gradient flow dynamics can be inferred from

$$\lambda_1 \geq \frac{\varepsilon_1}{\prod_{i=2}^n \lambda_i},$$

by picking the largest possible $\prod_{i=2}^n \lambda_i$ given the constraint $\sum_{i=2}^n \lambda_i \leq n$. This is achieved by taking

$$\lambda_i = \frac{n}{n-1}, \quad \forall i \in \{2, \cdots, n\},$$

which gives

$$\prod_{i=2}^{n} \lambda_i = \left(\frac{n}{n-1} \right)^{n-1} = \left(1 + \frac{1}{n-1} \right)^{n-1} \leq C,$$

where C is a universal constant, since the RHS converges from below to Euler's number as n increases. This proves that λ_1 is bounded away from zero uniformly in t. As a result, we can readily conclude that (C.46) holds. To see this last claim, consider a vector $\boldsymbol{v}$ which has 1 on position i and $-\mathrm{sign}\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle$ on position j. Hence, we have that

$$2\lambda_1 = \lambda_1 \cdot \|\boldsymbol{v}\|_2^2 \leq \boldsymbol{v}^\top(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top)\boldsymbol{v} = 2 - 2\cdot |\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle| \Rightarrow |\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle| \leq 1 - \lambda_1.$$

Notice that

$$\phi(t) \leq (n^2 - n) f(1),$$

since $xf(x) \leq f(1)$ for $|x| \leq 1$. Hence, we have that $\beta(t) \geq \frac{1}{nf(1)} > 0$. In this view, along the trajectory of the gradient flow (C.45), the quantity $\phi(t)$ is *strictly* decreasing until convergence, by the property of gradient flow.

Lemma C.3.1 (Characterization of stationary points)**.** *Consider the gradient flow* (C.45)*. Then, the following holds:*

(A) Any orthogonal set of b_i is a stationary point and a global minimizer.

(B) The gradient flow (C.45) *never escapes any subspace spanned by a set of linearly dependent b_i. However, for each such subspace there exists a direction in which* (C.43) *can be improved.*

Proof of Lemma C.3.1. Recall that $\beta(t) > 0$ and $\{\boldsymbol{b}_i\}_{i=1}^n$. Then, the stationary point condition can be expressed as

$$\boldsymbol{J}_k \sum_{j \neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle\right) = 0, \quad \forall k \in [n]. \tag{C.48}$$

Thus, any orthogonal set of vectors is clearly a stationary point by definition of $g(\cdot)$. Moreover, (C.43) is minimized *iff* $\boldsymbol{B}\boldsymbol{B}^\top = \boldsymbol{I}$ as $xf(x)$ is an even function since $f(\cdot)$ is odd.

Note that the kernel of the operator $\boldsymbol{J}_k$ is spanned by the vector $\boldsymbol{b}_k$. Thus, the condition (C.48) is equivalent to

$$\sum_{j \neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle\right) = \gamma_k \cdot \boldsymbol{b}_k,$$

for some $\gamma_k \in \mathbb{R}$. One can readily verify that $g(x) = 0$ if and only if $x = 0$. Thus, either *(i)* $\boldsymbol{b}_k$ is orthogonal to $\boldsymbol{b}_j$ for all $j \neq k$ and $\gamma_k = 0$, or *(ii)* $\boldsymbol{b}_k$ lies in the span of $\{\boldsymbol{b}_j\}_{j\neq k}$. If condition *(i)* holds for all $k \in [n]$, then $\{\boldsymbol{b}_i\}_{i=1}^n$ form an orthogonal set of vectors and we fall in category (A). If condition *(ii)* holds for some $k \in [n]$, then we fall in category (B).

Now, let us show that, if $\{\boldsymbol{b}_i\}_{i=1}^n$ spans a sub-space of dimension smaller than n, then there is a direction along which the value of (C.43) can be improved. Since the $\{\boldsymbol{b}_i\}_{i=1}^n$ are linearly dependent, there exists $\boldsymbol{u}$ of unit norm such that

$$\langle \boldsymbol{u}, \boldsymbol{b}_j\rangle = 0, \quad \forall j \in [n]. \tag{C.49}$$

For some $k \in [n]$, consider the perturbation

$$\hat{\boldsymbol{b}}_k = \frac{1}{\sqrt{1+\lambda^2}} \cdot (\boldsymbol{b}_k + \lambda \cdot \boldsymbol{u}),$$

which has unit norm as $\langle \boldsymbol{b}_k, \boldsymbol{u}\rangle = 0$. Recall that (C.43) can be expressed as

$$\beta^2 \left(2 \cdot \sum_{j\neq k}^{n} \left\langle \hat{\boldsymbol{b}}_k, \boldsymbol{b}_j \right\rangle f\left(\left\langle \hat{\boldsymbol{b}}_k, \boldsymbol{b}_j \right\rangle\right) + \frac{\pi}{2} + \sum_{i,j\neq k}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle f\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j\rangle\right)\right) - 2\beta n. \tag{C.50}$$

Here, β is chosen to be the minimizer of the quantity (C.50) having fixed $\{\boldsymbol{b}_j\}_{j\neq k}$ and $\hat{\boldsymbol{b}}_k$. Thus, in order to prove that the population risk gets smaller by replacing $\boldsymbol{b}_k$ with $\hat{\boldsymbol{b}}_k$ for any $\lambda > 0$, it suffices to show that the following quantity

$$\sum_{j\neq k}^{n} \left\langle \hat{\boldsymbol{b}}_k, \boldsymbol{b}_j \right\rangle f\left(\left\langle \hat{\boldsymbol{b}}_k, \boldsymbol{b}_j \right\rangle\right), \tag{C.51}$$

is decreasing with λ. This last claim follows from the chain of inequalities below:

$$\begin{aligned}
\text{(C.51)} &= \frac{1}{\sqrt{1+\lambda^2}} \sum_{j\neq k} \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle \cdot f\left(\frac{1}{\sqrt{1+\lambda^2}}\langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle\right) && \text{(C.52)}\\
&= \frac{1}{\sqrt{1+\lambda^2}} \sum_{j\neq k} \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle \cdot \sum_{\ell=0}^{\infty} \left(\frac{c_{2\ell+1}}{c_1}\right)^2 \cdot \left(\frac{1}{\sqrt{1+\lambda^2}}\right)^{2\ell+1} \cdot \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle^{2\ell+1} && \text{(C.53)}\\
&\leq \left(\frac{1}{\sqrt{1+\lambda^2}}\right)^2 \sum_{j\neq k} \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle \cdot \sum_{\ell=0}^{\infty} \left(\frac{c_{2\ell+1}}{c_1}\right)^2 \cdot \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle^{2\ell+1} && \text{(C.54)}\\
&= \frac{1}{1+\lambda^2} \sum_{j\neq k} \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle \cdot f\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle\right) < \sum_{j\neq k} \langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle \cdot f\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle\right), && \text{(C.55)}
\end{aligned}$$

where in the second line we substitute the Taylor expansion of $f(\cdot)$, the inequality in the third line uses that the coefficients $\{c_{2\ell+1}^2\}_{\ell=0}^{\infty}$ are all non-negative, and the last inequality follows from the fact that $\lambda > 0$.

Finally, we show that the gradient flow (C.45) does not escape the degenerate sub-space. If $\dim(\text{span}(\{\boldsymbol{b}_i\}_{i=1}^n)) < n$, then there exists $\boldsymbol{u} \in \mathbb{R}^d$ such that (C.49) holds. By projecting the gradient expression (C.48) onto $\boldsymbol{u}$, we have

$$\left\langle \boldsymbol{u}, \boldsymbol{J}_k \sum_{j\neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j\rangle\right)\right\rangle = 0.$$

Hence, for any $k \in [n]$, the directional derivative of $\boldsymbol{b}_k$ in the direction of $\boldsymbol{u}$ is equal to zero, and the gradient flow does not escape the low-rank sub-space, which concludes the proof. □

In next lemma we show that, if at initialization $\{\boldsymbol{b}_i\}_{i=1}^n$ spans a sub-space of dimension n, then it will never get stuck in a low-rank sub-space.

Lemma C.3.2 (Linearly independent $\{\boldsymbol{b}_i\}_{i=1}^n$ stay linearly independent)**.** *Consider the gradient flow* (C.45) *with full rank initialization, i.e.,* $\text{rank}(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) = n$. *Then, the following holds*

$$\frac{\partial}{\partial t} \log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) \geq 2\beta(t)^2 \cdot \phi(t) \geq 0,$$

where $\boldsymbol{B}(t)^\top = [\boldsymbol{b}_1(t), \cdots, \boldsymbol{b}_n(t)]$ *and* $\phi(t)$ *is defined in* (C.44)*. In particular, this implies that* $\{\boldsymbol{b}_i\}_{i=1}^n$ *stay full-rank along the gradient flow trajectory.*

Proof of Lemma C.3.2. Applying the chain rule and using that the time derivative of $\boldsymbol{B}$ is given by the gradient flow (C.45) implies that

$$\frac{\partial}{\partial t} \log\det(\boldsymbol{B}\boldsymbol{B}^\top) = \text{Tr}\left[(\boldsymbol{B}\boldsymbol{B}^\top)^{-1} \cdot \left(\frac{\partial \boldsymbol{B}}{\partial t} \cdot \boldsymbol{B}^\top + \boldsymbol{B} \cdot \frac{\partial \boldsymbol{B}^\top}{\partial t}\right)\right],$$

where

$$\frac{\partial \boldsymbol{b}_k}{\partial t} = -\beta(t)^2 \cdot \left(\boldsymbol{J}_k \sum_{j \neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle\right) \right).$$

Let us compute the quantity

$$\left\langle \frac{\partial \boldsymbol{b}_k}{\partial t}, \boldsymbol{b}_\ell \right\rangle = \left(\frac{\partial \boldsymbol{B}}{\partial t} \cdot \boldsymbol{B}^\top \right)_{k,\ell}.$$

By definition of $\boldsymbol{J}_k$, we have that

$$\boldsymbol{J}_k \sum_{j \neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle\right) = \sum_{j \neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle\right) - \sum_{j \neq k} \boldsymbol{b}_k \cdot \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle\right).$$

Note that

$$\left\langle \sum_{j \neq k} \boldsymbol{b}_k \cdot \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle\right), \boldsymbol{b}_\ell \right\rangle = \left[\mathrm{Diag}\left[\mathbf{1}^\top ((\boldsymbol{B}\boldsymbol{B}^\top - \boldsymbol{I}) \circ g(\boldsymbol{B}\boldsymbol{B}^\top)) \right] \cdot \boldsymbol{B}\boldsymbol{B}^\top \right]_{k,\ell},$$

and that

$$\left\langle \sum_{j \neq k} \boldsymbol{b}_j \cdot g\left(\langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle\right), \boldsymbol{b}_\ell \right\rangle = \left[g(\boldsymbol{B}\boldsymbol{B}^\top) \cdot \boldsymbol{B}\boldsymbol{B}^\top \right]_{k,\ell} - g(1) \cdot [\boldsymbol{B}\boldsymbol{B}^\top]_{k,\ell}.$$

By combining these last four equations, we conclude that

$$\begin{aligned}\frac{\partial \boldsymbol{B}}{\partial t} \cdot \boldsymbol{B}^\top = -\beta(t)^2 \Big(& g(\boldsymbol{B}\boldsymbol{B}^\top) \cdot \boldsymbol{B}\boldsymbol{B}^\top - g(1) \cdot \boldsymbol{B}\boldsymbol{B}^\top \\ & - \mathrm{Diag}\left[\mathbf{1}^\top ((\boldsymbol{B}\boldsymbol{B}^\top - \boldsymbol{I}) \circ g(\boldsymbol{B}\boldsymbol{B}^\top)) \right] \cdot \boldsymbol{B}\boldsymbol{B}^\top \Big).\end{aligned}$$

Furthermore,

$$\begin{aligned}\boldsymbol{B} \cdot \frac{\partial \boldsymbol{B}^\top}{\partial t} = \left(\frac{\partial \boldsymbol{B}}{\partial t} \cdot \boldsymbol{B}^\top \right)^\top = -\beta(t)^2 \Big(& \boldsymbol{B}\boldsymbol{B}^\top \cdot g(\boldsymbol{B}\boldsymbol{B}^\top) - g(1) \cdot \boldsymbol{B}\boldsymbol{B}^\top \\ & - \boldsymbol{B}\boldsymbol{B}^\top \cdot \mathrm{Diag}\left[\mathbf{1}^\top ((\boldsymbol{B}\boldsymbol{B}^\top - \boldsymbol{I}) \circ g(\boldsymbol{B}\boldsymbol{B}^\top)) \right] \Big).\end{aligned}$$

Hence, by using the cyclic property of the trace, we get that

$$\begin{aligned}\frac{\partial}{\partial t} \log\det(\boldsymbol{B}\boldsymbol{B}^\top) &= 2\beta(t)^2 \cdot \mathrm{Tr}\left[\mathrm{Diag}\left[\mathbf{1}^\top ((\boldsymbol{B}\boldsymbol{B}^\top - \boldsymbol{I}) \circ g(\boldsymbol{B}\boldsymbol{B}^\top)) \right] \right] \\ &\quad - 2\beta(t)^2 \cdot \mathrm{Tr}\left[g(\boldsymbol{B}\boldsymbol{B}^\top) - g(1) \cdot \boldsymbol{I} \right] \\ &= 2\beta(t)^2 \cdot \sum_{i \neq j}^{n} \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \cdot g\left(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle\right) + 0,\end{aligned}$$

Now, note that

$$xg(x) = x^2 f'(x) + x f(x) \geq 0,$$

since $x^2 f'(x)$ and $x f(x)$ are non-negative functions, which concludes the proof. □

The result of Lemma C.3.2 gives that $\det(\boldsymbol{B}\boldsymbol{B}^\top)$ is non-decreasing. Hence, if $\lambda_{\min}(\boldsymbol{B}\boldsymbol{B}^\top) > \delta > 0$ at initialization, then this quantity will be bounded away from zero during the gradient flow dynamics and the gradient flow will not get stuck in a low-rank solution. Therefore, by Lemma C.3.1, the gradient flow converges to a global minimum, in which the rows of $\boldsymbol{B}$ are orthogonal vectors with unit norm. The speed at which this happens is characterized by the next lemma.

Lemma C.3.3 (Rate of convergence)**.** *Consider the gradient flow* (C.45) *with full rank initialization, i.e.,* $\operatorname{rank}(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) = n$*. Let* T *be the time at which* $\phi(T)$ *hits the value* $\delta > 0$*. Then, the following holds*

$$T \leq -\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) \cdot \left(f(1) \cdot \mathbb{1}\{\phi(0) > n \cdot f(1)\} + \frac{2f^2(1)}{\delta} \cdot \mathbb{1}\{\delta \leq n \cdot f(1)\} \right). \tag{C.56}$$

Proof of Lemma C.3.3. For all t, we have that $\operatorname{Tr}\left[\boldsymbol{B}(t)\boldsymbol{B}(t)^\top\right] = n$, which implies that $\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) \leq 1$ and, as a consequence, that $\log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) \leq 0$. From Lemma C.3.2, we know that

$$\frac{\partial}{\partial t} \log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) \geq 2\beta(t)^2 \cdot \phi(t).$$

In this view, using the exact expression (C.45) for $\beta(t)$, we get

$$\begin{aligned} -\log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) &\geq \log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) - \log\det(\boldsymbol{B}(t)\boldsymbol{B}(t)^\top) \\ &\geq \int_0^t \frac{2}{\left(f(1) + \frac{\phi(s)}{n}\right)^2} \cdot \phi(s) \mathrm{d}s. \end{aligned} \tag{C.57}$$

Stage 1. Assume that $\phi(0) > n \cdot f(1)$, and let T_1 be such that $\phi(T_1) = n \cdot f(1)$. Recall that the function $\phi(t)$ is decreasing and note that $x/(1+x)^2$ is decreasing for $x \in [1, +\infty)$. In this view, we can lower bound the integrand in the RHS of (C.57) for all $t \leq T_1$ by

$$\frac{2 \cdot \phi(0)}{\left(f(1) + \frac{\phi(0)}{n}\right)^2} \geq \frac{2(n-1)}{nf(1)} \geq \frac{1}{f(1)}, \tag{C.58}$$

where the first inequality follows from the definition (C.44) of $\phi(\cdot)$, which readily implies that $\phi(0) \leq f(1) \cdot n(n-1)$. Hence, by combining (C.57) with the lower bound (C.58), we get

$$T_1 \leq -f(1) \cdot \log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top).$$

Stage 2. Assume that $\phi(0) \leq n \cdot f(1)$. Let $\delta \in (0, n \cdot f(1)]$ be the desired precision which should be reached during the gradient flow, and let T_2 be such that $\phi(T_2) = \delta$. As $\phi(t)$ is decreasing, we have that

$$\frac{1}{\left(f(1) + \frac{\phi(t)}{n}\right)^2} \geq \frac{1}{\left(f(1) + \frac{\phi(0)}{n}\right)^2} \geq \frac{1}{4f^2(1)}, \tag{C.59}$$

where in the last step we use that $\phi(0) \leq n \cdot f(1)$. Hence, by combining (C.57) with the lower bound (C.59), we get

$$-\log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top) \geq \frac{1}{2f^2(1)} \cdot T_2\delta,$$

which implies that

$$T_2 \leq -\frac{2f^2(1) \cdot \log\det(\boldsymbol{B}(0)\boldsymbol{B}(0)^\top)}{\delta}.$$

By combining the results of both stages, the desired result (C.56) readily follows. □

Proof of Theorem 11. Theorem 11 is a compilation of the results presented in current section. □

C.4 Global Convergence of Projected Gradient Descent (Theorem 7)

Recall from statement of Theorem 7 that

$$f(x) = x + \sum_{\ell=3}^{\infty} c_\ell^2 x^\ell,$$

with $\sum_{\ell=3}^{\infty} c_\ell^2 < \infty$. We also define $\alpha = \sum_{\ell=3}^{\infty} c_\ell^2$, and we assume that $\alpha > 0$. In fact, if $\alpha = 0$, then the algorithm trivially converges after one step. We denote by C, c uniform positive constants (depending only on r and α) the value of which might change from term to term. To make the notation lighter we will also but the time t as a subscript (for example $\boldsymbol{B}(t)$ becomes $\boldsymbol{B}_t$).

We analyze the following projected gradient descent procedure for minimizing the population risk

$$\sum_{i,j=1}^{n} \langle \boldsymbol{a}_i, \boldsymbol{a}_j \rangle \cdot f\left(\left\langle \frac{\boldsymbol{b}_i}{\|\boldsymbol{b}_i\|_2}, \frac{\boldsymbol{b}_j}{\|\boldsymbol{b}_j\|_2} \right\rangle\right) - 2\sum_{i=1}^{n} \left\langle \boldsymbol{a}_i, \frac{\boldsymbol{b}_i}{\|\boldsymbol{b}_i\|_2} \right\rangle. \tag{C.60}$$

Given unit-norm initial $\{\boldsymbol{b}_i\}_{i\in[n]}$, at each step we pick the optimal value of $\boldsymbol{A}$ given $\boldsymbol{B}$

$$\boldsymbol{A}_t = \boldsymbol{B}_t^\top \left(f(\boldsymbol{B}_t \boldsymbol{B}_t^\top)\right)^{-1}. \tag{C.61}$$

Then, we update $\boldsymbol{B}_t$ with a gradient step and a projection on the sphere to keep the unit norm:

$$\boldsymbol{B}_t' := \boldsymbol{B}_t - \eta \nabla_{\boldsymbol{B}_t}, \quad \boldsymbol{B}_{t+1} := \mathrm{proj}(\boldsymbol{B}_t').$$

Here, the operator $\mathrm{proj}(\boldsymbol{M})$ normalizes the rows of $\boldsymbol{M}$ to be of unit norm and each row of $\nabla_{\boldsymbol{B}_t}$ is defined as the corresponding row of the gradient of $\boldsymbol{B}_t$, i.e.,

$$(\nabla_{\boldsymbol{B}_t})_{k,:} = \underbrace{-2\boldsymbol{J}_k \boldsymbol{a}_k + 2\sum_{j\neq k} \langle \boldsymbol{a}_k, \boldsymbol{a}_j \rangle \boldsymbol{J}_k \boldsymbol{b}_j}_{:=\nabla^1_{\boldsymbol{B}_t} \quad \text{(part 1)}} + \sum_{l=3}^{\infty} \underbrace{\ell c_\ell^2 \sum_{j\neq k} \langle \boldsymbol{a}_k, \boldsymbol{a}_j \rangle \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle^{l-1} \boldsymbol{J}_k \boldsymbol{b}_j}_{:=\nabla^2_{\boldsymbol{B}_t} \quad \text{(part 2)}}, \tag{C.62}$$

where $\boldsymbol{J}_k := \boldsymbol{I} - \boldsymbol{b}_k \boldsymbol{b}_k^\top$ and we have omitted the iteration number t on $\{\boldsymbol{a}_j, \boldsymbol{b}_j\}_{j\in[n]}$ to keep notation light. Note that in (C.62) the norms $\|\boldsymbol{b}_i\|_2$, $\|\boldsymbol{b}_j\|_2$ no longer appear as the projection step enforces $\|\boldsymbol{b}_i\|_2 = 1$. At each step of the projected gradient descent dynamics, we decompose $\boldsymbol{B}_t \boldsymbol{B}_t^\top$ as follows:

$$\boldsymbol{B}_t \boldsymbol{B}_t^\top = \boldsymbol{I} + \boldsymbol{Z}_t + \boldsymbol{X}_t, \tag{C.63}$$

where $\boldsymbol{B}_0 \boldsymbol{B}_0^\top = \boldsymbol{U} \boldsymbol{\Lambda}_0 \boldsymbol{U}^\top$, $\boldsymbol{Z}_t = \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top$ and $\boldsymbol{\Lambda}_{t+1} = g(\boldsymbol{\Lambda}_t)$ for some function $g : \mathbb{R}^n \to \mathbb{R}^n$ which defines the spectrum evolution. Here, $\boldsymbol{U}$ is an orthogonal matrix that importantly does not depend on t and $\boldsymbol{\Lambda}_t$ is the diagonal matrix containing the eigenvalues (i.e., $\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top$ is the SVD). We also define $\boldsymbol{X}_t^D := \mathrm{Diag}(\boldsymbol{X}_t)$ and $\boldsymbol{X}_t^O := \boldsymbol{X}_t - \boldsymbol{X}_t^D$.

For now we will make the following assumptions, which will be proved later in the argument. There exist universal constants $C, C_X > 0$ and $\delta \in (0,1)$ (depending only on r) such that, with probability at least $1 - Ce^{-cd}$,

$$\begin{aligned}
&\inf_{t\geq 0} \lambda_{\min}(\boldsymbol{Z}_t) \geq -1 + \delta_r, \\
&\sup_{t\geq 0} \|\boldsymbol{Z}_t\|_{op} \leq C, \\
&\sup_{t\geq 0} \|\boldsymbol{X}_t\|_{op} \leq C_X \frac{\mathrm{poly}(\log d)}{\sqrt{d}}, \\
&\|\boldsymbol{\Lambda}_t - \boldsymbol{I}\|_{op} \leq C\, e^{-c\eta t}.
\end{aligned} \tag{C.64}$$

Here, $\mathrm{poly}(\log d)$ is used to denote polynomial powers of $\log d$, i.e., $(\log d)^C$ for some universal constant C. In the assumptions (C.64), we specifically distinguish the constant C_X in the bound on $\|\boldsymbol{X}_t\|_{op}$ from the others. This important distinction between C and C_X will be apparent later to show that assumptions (C.64) indeed hold. Note also that, for sufficiently large d, (C.64) implies that

$$\sup_{t\geq 0} \|\boldsymbol{X}_t\|_{op} \leq 1. \tag{C.65}$$

We are now ready to give the proof Theorem 7. For the convenience of the reader we restate it here.

Theorem 12. *Consider the projected gradient descent algorithm as described above applied to the objective* (5.13) *for any f of the form $f(x) = x + \sum_{\ell=3} c_\ell^2 x^\ell$, where $\sum_{\ell=3} c_\ell^2 < \infty$. Initialize the algorithm with $\boldsymbol{B}_0$ equal to a row-normalized Gaussian, i.e., $(\boldsymbol{B}_0')_{i,j} \sim \mathcal{N}(0, 1/d)$, $(\boldsymbol{B}_0)i,: = \mathbf{Proj}_{\mathbb{S}^{d-1}}\left((\boldsymbol{B}_0')_{i,:}\right)$. Let the step size η be $\Theta(1/\sqrt{d})$. Then, for any $r < 1$, we have that at any time $t = T/\eta$, with probability at least $1 - Ce^{-cd}$,*

$$\left\|\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I}\right\|_{op} \leq C(1-c)^T,$$

where $C > 0$ and $c \in (0, 1]$ are universal constants depending only on r and f.

Let $\boldsymbol{E}^t := \boldsymbol{E}(\boldsymbol{X}_t, \boldsymbol{Z}_t) \in \mathbb{R}^{n\times n}$ be a generic matrix whose operator norm is upper bounded by

$$\left\|\boldsymbol{E}^t\right\|_{op} \leq C\left(\frac{\mathrm{poly}(\log d)}{\sqrt{d}} \cdot \|\boldsymbol{Z}_t\|_{op}^{1/2} + \|\boldsymbol{X}_t\|_{op}^2 + \|\boldsymbol{X}_t\|_{op}\|\boldsymbol{Z}_t\|_{op}^{1/2}\right). \tag{C.66}$$

We highlight that the constant in front of the upper-bound on the error term $\boldsymbol{E}^t$ is *independent* of C_X and t.

Lemma C.4.1 (Bound for the matrix inverse)**.** *Assume that* (C.64) *holds. Then, for all sufficiently large n, with probability at least $1 - 1/d^2$, jointly for all $t \geq 0$ and $\ell \geq 3$, the following bounds hold*

$$\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ\ell}\|_{op} \leq \|\boldsymbol{E}^t\|_{op}, \tag{C.67}$$

$$\|\left(f(\boldsymbol{B}_t\boldsymbol{B}_t^\top)\right)^{-1} - (\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op} \leq \|\boldsymbol{E}^t\|_{op}, \tag{C.68}$$

where α was defined as $\alpha = \sum_{\ell=3}^{\infty} c_\ell^2$.

Proof of Lemma C.4.1. Note that, for any square matrices $\boldsymbol{R}, \boldsymbol{S} \in \mathbb{R}^{n\times n}$,

$$\|\boldsymbol{R} \circ \boldsymbol{S}\|_{op} \leq \sqrt{n}\|\boldsymbol{S}\|_{op} \max_{i,j} |\boldsymbol{R}_{i,j}|. \tag{C.69}$$

Thus, for $\ell \geq 3$,

$$\begin{aligned} \left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ\ell}\right\|_{op} &\leq \sqrt{n}\left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op} \max_{i,j} |((\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3})_{i,j}| \\ &= \sqrt{n}\left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op} \max_{i\neq j} |((\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3})_{i,j}| \\ &= \sqrt{n}\left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op} \max_{i\neq j} |((\boldsymbol{Z}_t + \boldsymbol{X}_t)^{\circ 3})_{i,j}|, \end{aligned} \tag{C.70}$$

where in the first line we use (C.69), in the second line we use that $((\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3})_{i,i} = 0$ for $i \in [n]$ and in the third line we use the decomposition (C.63).

Let us bound the off-diagonal entries of $\boldsymbol{X}_t$ via (C.64) and the off-diagonal entries of $\boldsymbol{Z}_t$ via Lemma C.5.2. This gives that, with probability at least $1 - 1/d^2$, jointly for all $t \geq 0$,

$$\max_{i\neq j} |((\boldsymbol{Z}_t + \boldsymbol{X}_t)^{\circ 3})_{i,j}| \leq (C + C_X)^3 \left(\frac{\mathrm{poly}(\log d)}{d}\right)^{3/2}. \tag{C.71}$$

We will condition on this event (without explicitly mentioning it every time) for the reminder of the argument. By combining (C.70) and (C.71), we have that

$$\begin{aligned} \left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ\ell}\right\|_{op} &\leq \sqrt{n}\left[(C + C_X)^3 \left(\frac{\mathrm{poly}(\log d)}{d}\right)^{3/2}\right] \left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op} \\ &\leq \left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op} \end{aligned} \tag{C.72}$$

where the last inequality holds for all sufficiently large n. Note that, for any square matrices R, S, an application of Theorem 1 in [Vis00] gives that

$$\|\boldsymbol{R} \circ \boldsymbol{S}\|_{op} \leq \|\boldsymbol{R}\|_{op}\|\boldsymbol{S}\|_{op}. \tag{C.73}$$

Hence,

$$\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ\ell}\|_{op} \leq \left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op} \|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3}\|_{op}. \tag{C.74}$$

Now, by using again (C.73) and the assumptions (C.64), we have that, for $\ell \in [3]$,

$$\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ\ell}\|_{op} \leq C. \tag{C.75}$$

Thus, by combining (C.72) and (C.75), we obtain that $\left\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-3)}\right\|_{op}$ is uniformly bounded in ℓ, which together with (C.74) gives that

$$\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ\ell}\|_{op} \leq C\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3}\|_{op}. \tag{C.76}$$

We remark here that C is independent of l and C_X. This means that it suffices to prove the claim (C.67) for $\ell = 3$.

To do so, define $\boldsymbol{H} := \boldsymbol{1}\boldsymbol{1}^\top - \boldsymbol{I}$, hence, since $\boldsymbol{B}_t\boldsymbol{B}_t^\top$ has unit diagonal, we have that

$$\begin{aligned} (\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3} &= (\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 3} \circ \boldsymbol{H} = (\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top + \boldsymbol{X}_t^O + \boldsymbol{X}_t^D)^{\circ 3} \circ \boldsymbol{H} \\ &= (\boldsymbol{Z}_t \circ \boldsymbol{H} + \boldsymbol{X}_t^O \circ \boldsymbol{H} + \boldsymbol{X}_t^D \circ \boldsymbol{H})^{\circ 3} = (\boldsymbol{Z}_t \circ \boldsymbol{H} + \boldsymbol{X}_t^O)^{\circ 3} \\ &= (\boldsymbol{Z}_t \circ \boldsymbol{H})^{\circ 3} + 3(\boldsymbol{Z}_t \circ \boldsymbol{H})^{\circ 2} \circ \boldsymbol{X}_t^O + 3(\boldsymbol{Z}_t \circ \boldsymbol{H}) \circ (\boldsymbol{X}_t^O)^{\circ 2} + (\boldsymbol{X}_t^O)^{\circ 3}. \end{aligned}$$

Using again (C.73) and that, by Lemma C.5.1 for any $\boldsymbol{R} \in \mathbb{R}^{n\times n}$,

$$\|\boldsymbol{R} \circ \boldsymbol{H}\|_{op} = \|\boldsymbol{R} - \mathrm{diag}(\boldsymbol{R})\|_{op} \leq C\|\boldsymbol{R}\|_{op},$$

we get

$$\begin{aligned}\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top-\boldsymbol{I})^{\circ 3}\|_{op}&\le C\left(\|(\boldsymbol{Z}_t\circ\boldsymbol{H})^{\circ 3}\|_{op}+\|\boldsymbol{Z}_t\|_{op}^2\|\boldsymbol{X}_t^O\|_{op}+\|\boldsymbol{Z}_t\|_{op}\|\boldsymbol{X}_t^O\|_{op}^2+\|\boldsymbol{X}_t^O\|_{op}^3\right)\\&\le C\left(\|(\boldsymbol{Z}_t\circ\boldsymbol{H})^{\circ 3}\|_{op}+\|\boldsymbol{Z}_t\|_{op}^{1/2}\|\boldsymbol{X}_t^O\|_{op}+\|\boldsymbol{X}_t^O\|_{op}^2\right),\end{aligned}\tag{C.77}$$

where the second step holds since $\left\|\boldsymbol{X}_t^O\right\|_{op}\le 1$ and $\|\boldsymbol{Z}_t\|_{op}\le C$ by (C.64)-(C.65). Another application of (C.69) gives that

$$\begin{aligned}\|(\boldsymbol{Z}_t\circ\boldsymbol{H})^{\circ 3}\|_{op}=\|(\boldsymbol{Z}_t\circ\boldsymbol{H})^{\circ 2}\circ\boldsymbol{Z}_t\|_{op}&\le\sqrt{n}\cdot\max_{i\neq j}|(\boldsymbol{Z}_t)_{i,j}|^2\cdot\|\boldsymbol{Z}_t\|_{op}\\&\le C\frac{\log d}{\sqrt{d}}\cdot\|\boldsymbol{Z}_t\|_{op}\le C\frac{\log d}{\sqrt{d}}\cdot\|\boldsymbol{Z}_t\|_{op}^{1/2},\end{aligned}\tag{C.78}$$

where the second passage follows from Lemma C.5.2 and the last from $\|\boldsymbol{Z}_t\|_{op}\le C$. By combining (C.77) and (C.78), the proof of (C.67) for $\ell=3$ is complete.

To prove (C.68), define the following quantity

$$\boldsymbol{Y}:=\sum_{\ell=3}^{\infty}c_\ell^2(\boldsymbol{B}_t\boldsymbol{B}_t^\top-\boldsymbol{I})^{\circ\ell}.$$

By definition of $f(\cdot)$ we have that

$$f(\boldsymbol{B}_t\boldsymbol{B}_t^\top)=\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top+\boldsymbol{Y},$$

which implies that

$$\begin{aligned}\left(f(\boldsymbol{B}_t\boldsymbol{B}_t^\top)\right)^{-1}&=(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top+\boldsymbol{Y})^{-1}\\&=(\boldsymbol{I}+\boldsymbol{Y}(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1})^{-1}(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\\&=\left(\boldsymbol{I}+\sum_{k=1}^{\infty}(-1)^k(\boldsymbol{Y}(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1})^k\right)(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}.\end{aligned}\tag{C.79}$$

By definition (C.66), we have that $\|\boldsymbol{E}^t\|_{op}\le 1/2$ under assumptions (C.64) for sufficiently large d. Hence, by the result (C.67) we have just proved, $\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top-\boldsymbol{I})^{\circ\ell}\|_{op}\le 1/2$, which implies that $\sum_{\ell=3}^{\infty}c_\ell^2\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top-\boldsymbol{I})^{\circ\ell}\|_{op}\le\alpha/2$. Thus, we have

$$\|\boldsymbol{Y}(\boldsymbol{B}_t\boldsymbol{B}_t^\top+\alpha\boldsymbol{I})^{-1}\|_{op}\le\|\boldsymbol{Y}\|_{op}\|(\boldsymbol{B}_t\boldsymbol{B}_t^\top+\alpha\boldsymbol{I})^{-1}\|_{op}\le\frac{\alpha}{2}\cdot\frac{1}{\alpha}\le\frac{1}{2}.\tag{C.80}$$

Therefore, we can conclude that

$$\begin{aligned}\|\left(f(\boldsymbol{B}_t\boldsymbol{B}_t^\top)\right)^{-1}-(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op}&\le\|(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op}\cdot\sum_{k=1}^{\infty}\|\boldsymbol{Y}(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op}^k\\&\le\frac{1}{\alpha}\cdot\frac{\|\boldsymbol{Y}(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op}}{1-\|\boldsymbol{Y}(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op}}\\&\le\frac{2}{\alpha}\cdot\|\boldsymbol{Y}\|_{op}\|(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op}\\&\le\frac{2}{\alpha^2}\cdot\|\boldsymbol{Y}\|_{op},\end{aligned}\tag{C.81}$$

where the third inequality uses (C.80). By bounding $\|\boldsymbol{Y}\|_{op}$ via (C.67), the proof of (C.68) is complete. □

Lemma C.4.2 (Bound for the Schur product with $\boldsymbol{A}^\top \boldsymbol{A}$)**.** *Assume that* (C.64) *holds, and let* $\boldsymbol{A}_t$ *be given by* (C.61)*. Then, we have that, with probability at least* $1 - 1/d^2$*, jointly for all* $t \geq 0$ *and* $\ell \geq 2$*,*

$$\left\| \boldsymbol{A}_t^\top \boldsymbol{A}_t \circ (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell} \right\|_{op} \leq \|\boldsymbol{E}^t\|_{op}. \tag{C.82}$$

Proof of Lemma C.4.2. We have that

$$\begin{aligned} \|\boldsymbol{A}_t^\top \boldsymbol{A}_t \circ (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}\|_{op} &\leq \|\boldsymbol{A}_t^\top \boldsymbol{A}_t \circ (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 2}\|_{op} \left\| (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ (\ell - 2)} \right\|_{op} \\ &\leq C \|\boldsymbol{A}_t^\top \boldsymbol{A}_t \circ (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 2}\|_{op}, \end{aligned} \tag{C.83}$$

where the first inequality uses (C.73) and the second inequality uses that $\left\| (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ (\ell - 2)} \right\|_{op}$ is uniformly bounded in l, which follows from (C.72) and (C.75).

Let us now focus on bounding the RHS of (C.83). An application of Lemma C.4.1 gives that

$$\left(f(\boldsymbol{B}_t \boldsymbol{B}_t^\top) \right)^{-1} = (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} + \boldsymbol{E}_1,$$

where

$$\|\boldsymbol{E}\|_{op} \leq \left\| \boldsymbol{E}^t \right\|_{op}.$$

Hence, by using (C.61), we get that

$$\begin{aligned} \boldsymbol{A}_t^\top \boldsymbol{A}_t &= ((\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} \boldsymbol{B}_t + \boldsymbol{E}_1^\top \boldsymbol{B}_t)(\boldsymbol{B}_t^\top (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} + \boldsymbol{B}_t^\top \boldsymbol{E}_1) \\ &= \boldsymbol{B}_t \boldsymbol{B}_t^\top (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-2} + \boldsymbol{E}_1^\top \boldsymbol{B}_t \boldsymbol{B}_t^\top (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} \\ &\quad + (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} \boldsymbol{B}_t \boldsymbol{B}_t^\top \boldsymbol{E}_1 + \boldsymbol{E}_1^\top \boldsymbol{B}_t \boldsymbol{B}_t^\top \boldsymbol{E}_1, \end{aligned} \tag{C.84}$$

where we rearranged the first term in (C.84) using that $\boldsymbol{B}_t \boldsymbol{B}_t^\top$ and $(\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1}$ commute. By using the assumptions (C.64), we have that

$$\|\boldsymbol{B}_t \boldsymbol{B}_t^\top\|_{op} \leq C, \qquad \|\boldsymbol{E}_1\|_{op} \leq 1/2, \qquad \|(\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1}\|_{op} \leq \frac{1}{\alpha}.$$

Hence, we can upper bound the operator norm of the last three terms in (C.84) as

$$\left\| \boldsymbol{E}_1^\top \boldsymbol{B}_t \boldsymbol{B}_t^\top (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} + (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-1} \boldsymbol{B}_t \boldsymbol{B}_t^\top \boldsymbol{E}_1 + \boldsymbol{E}_1^\top \boldsymbol{B}_t \boldsymbol{B}_t^\top \boldsymbol{E}_1 \right\|_{op} \leq C \|\boldsymbol{E}_1\|_{op}. \tag{C.85}$$

Let us now take a closer look at the first term in (C.84). Recall that

$$\boldsymbol{B}_t \boldsymbol{B}_t^\top = \boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top + \boldsymbol{X}_t.$$

As the operator norm is sub-multiplicative, we have that

$$\|\boldsymbol{X}_t \cdot (\alpha \boldsymbol{I} + \boldsymbol{B}_t \boldsymbol{B}_t^\top)^{-2}\|_{op} \leq C \|\boldsymbol{X}_t\|_{op}. \tag{C.86}$$

Furthermore,

$$\begin{aligned} &\boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top (\alpha \boldsymbol{I} + \boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top + \boldsymbol{X}_t)^{-2} \\ &= \boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top \left((\boldsymbol{I} + \boldsymbol{X}_t (\alpha \boldsymbol{I} + \boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top)^{-1})(\alpha \boldsymbol{I} + \boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top) \right)^{-2} \\ &= \boldsymbol{U} \boldsymbol{\Lambda}_t \boldsymbol{U}^\top \boldsymbol{T}_1^{-1} \boldsymbol{T}_2^{-1} \boldsymbol{T}_1^{-1} \boldsymbol{T}_2^{-1}, \end{aligned} \tag{C.87}$$

where we have defined

$$\boldsymbol{T}_1 = \alpha \boldsymbol{I} + \boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top, \qquad \boldsymbol{T}_2 = \boldsymbol{I} + \boldsymbol{X}_t(\alpha\boldsymbol{I} + \boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top)^{-1}.$$

By expanding $\boldsymbol{T}_2^{-1}$ as in (C.79)-(C.81), we get

$$\|\boldsymbol{T}_2^{-1} - \boldsymbol{I}\|_{op} \le C\|\boldsymbol{X}_t\|_{op},$$

or equivalently

$$\boldsymbol{T}_2^{-1} = \boldsymbol{I} + \boldsymbol{E}_2,$$

with $\|\boldsymbol{E}_2\|_{op} \le C\|\boldsymbol{X}_t\|_{op}$. In this view, looking at (C.87) we have

$$\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top\boldsymbol{T}_1^{-1}\boldsymbol{T}_2^{-1}\boldsymbol{T}_1^{-1}\boldsymbol{T}_2^{-1} = \boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{B}\boldsymbol{U}^\top\boldsymbol{T}_1^{-1}(\boldsymbol{I}+\boldsymbol{E}_2)\boldsymbol{T}_1^{-1}(\boldsymbol{I}+\boldsymbol{E}_2).$$

All the terms which involve $\boldsymbol{E}_2$ can be controlled. We provide the analysis for two terms of different nature, the rest follows from similar arguments. As $\|\boldsymbol{T}_1^{-1}\|_{op} \le 1/\alpha$ and $\|\boldsymbol{\Lambda}_t\|_{op} \le C$, we have that

$$\begin{aligned}
\|\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top\boldsymbol{T}_1^{-1}\boldsymbol{E}_2\boldsymbol{T}_1^{-1}\boldsymbol{E}_2\|_{op} &\le \|\boldsymbol{T}_1^{-1}\|_{op}^2\|\boldsymbol{E}_2\|_{op}^2 \le \frac{C}{\alpha^2}\|\boldsymbol{X}_t\|_{op}^2 \le \frac{C}{\alpha^2}\|\boldsymbol{X}_t\|_{op},\\
\|\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top\boldsymbol{T}_1^{-1}\boldsymbol{I}\boldsymbol{T}_1^{-1}\boldsymbol{E}_2\|_{op} &\le \|\boldsymbol{T}_1^{-1}\|_{op}^2\|\boldsymbol{E}_2\|_{op} \le \frac{C}{\alpha^2}\|\boldsymbol{X}_t\|_{op},
\end{aligned}$$

where we have also used that $\|\boldsymbol{X}_t\|_{op}$ is bounded via assumptions (C.64). Furthermore, a simple manipulation gives

$$\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top\boldsymbol{T}_1^{-2} = \boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top(\alpha\boldsymbol{I}+\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top)^{-2} = \boldsymbol{U}\boldsymbol{\Lambda}_t(\alpha\boldsymbol{I}+\boldsymbol{\Lambda}_t)^{-2}\boldsymbol{U}^\top = \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top,$$

where $\phi(x) = \frac{x}{(\alpha+x)^2}$. As a result,

$$\left\|\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top\boldsymbol{T}_1^{-1}\boldsymbol{T}_2^{-1}\boldsymbol{T}_1^{-1}\boldsymbol{T}_2^{-1} - \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top\right\|_{op} \le C\left\|\boldsymbol{X}_t\right\|_{op},$$

which implies that

$$\|\boldsymbol{B}_t\boldsymbol{B}_t^\top(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} - \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top\|_{op} \le C\|\boldsymbol{X}_t\|_{op}. \tag{C.88}$$

By combining (C.84), (C.85) and (C.88), we have that

$$\|\boldsymbol{A}_t^\top\boldsymbol{A}_t - \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top\|_{op} \le C\Big(\|\boldsymbol{X}_t\|_{op} + \left\|\boldsymbol{E}_1\right\|_{op}\Big). \tag{C.89}$$

At this point, we are ready to analyze the operator norm of $\|\boldsymbol{A}_t^\top\boldsymbol{A}_t \circ (\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 2}\|_{op}$:

$$\begin{aligned}
\boldsymbol{A}_t^\top\boldsymbol{A}_t \circ (\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 2} &= (\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top + \boldsymbol{E}_3) \circ (\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top + \boldsymbol{X}_t)^{\circ 2} \circ \boldsymbol{H}\\
&= (\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top + \boldsymbol{E}_3) \circ ((\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top)^{\circ 2} + \boldsymbol{X}_t^{\circ 2}\\
&\qquad + 2(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top) \circ \boldsymbol{X}_t) \circ \boldsymbol{H},
\end{aligned} \tag{C.90}$$

where we have defined $\boldsymbol{H} := \boldsymbol{1}\boldsymbol{1}^\top - \boldsymbol{I}$ and $\|\boldsymbol{E}_3\|_{op} \le C\Big(\|\boldsymbol{X}_t\|_{op} + \left\|\boldsymbol{E}_1\right\|_{op}\Big)$. We now decompose the quantity into three terms:

$$\boldsymbol{A}_t^\top\boldsymbol{A}_t \circ (\boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 2} = \boldsymbol{S}_1 + \boldsymbol{S}_2 + \boldsymbol{S}_3,$$

where

$$\begin{aligned}
\boldsymbol{S}_1 &= (\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top \circ \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top \circ \boldsymbol{H}) \circ \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top, \\
\boldsymbol{S}_2 &= \boldsymbol{H} \circ \boldsymbol{E}_3 \circ ((\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top)^{\circ 2} + \boldsymbol{X}_t^{\circ 2} + 2(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top) \circ \boldsymbol{X}_t), \\
\boldsymbol{S}_3 &= \boldsymbol{H} \circ \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top \circ (\boldsymbol{X}_t^{\circ 2} + 2(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top) \circ \boldsymbol{X}_t).
\end{aligned}$$

We proceed to bound each of these terms separately.

We start with $\boldsymbol{S}_1$. As $\phi(x)$ is differentiable for $x \geq 0$, the derivative of $\phi(x)$ is bounded for any compact interval $I \subseteq \mathbb{R}_+$. Hence, $\phi(x)$ is locally Lipschitz on I with Lipschitz constant $C_I > 0$, which implies that

$$|\phi(x) - \phi(1)| = \left|\phi(x) - \frac{1}{(1+\alpha)^2}\right| \leq C_I|x-1|.$$

By assumption (C.64), we have that $\boldsymbol{\Lambda}_t \succ 0$ and $\|\boldsymbol{\Lambda}_t\|_{op} \leq C$, hence

$$\left\|\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top - \frac{1}{(1+\alpha)^2}\boldsymbol{I}\right\|_{op} \leq C_I \cdot \|\boldsymbol{Z}_t\|_{op}. \tag{C.91}$$

Hence, an application of Lemma C.5.2 gives that, with probability at least $1 - 1/d^2$,

$$\sup_{t\geq 0} m\left(\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top - \frac{1}{(1+\alpha)^2}\boldsymbol{I}\right) \leq c\sqrt{\frac{\log d}{d}}, \tag{C.92}$$

where $c > 0$ is a universal constant. Another application of Lemma C.5.2 also gives that, with the same probability,

$$\sup_{t\geq 0} m\left(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top\right) \leq c\sqrt{\frac{\log d}{d}}. \tag{C.93}$$

As a result, we obtain the bound

$$\begin{aligned}
\|\boldsymbol{S}_1\|_{op} &= \|([\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top - 1/(1+\alpha)^2\boldsymbol{I}] \circ \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top \circ \boldsymbol{H}) \circ \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top\|_{op} \\
&\leq C\frac{\log d}{\sqrt{d}}\|\boldsymbol{Z}_t\|_{op}.
\end{aligned} \tag{C.94}$$

Here, the first equality is due to the fact that we are taking the Hadamard product with the matrix $\boldsymbol{H}$ which has 0 on the diagonal, hence we can add multiples of the identity to $\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top$; and the second inequality uses (C.69) with $\boldsymbol{R} = [\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top - 1/(1+\alpha)^2\boldsymbol{I}] \circ \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top \circ \boldsymbol{H}$ and $\boldsymbol{S} = \boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top$ in combination with (C.92)-(C.93).

Next, we bound $\|\boldsymbol{S}_2\|_{op}$. We inspect the terms appearing in the expression for $\boldsymbol{S}_2$ one by one. First note that we can omit $\boldsymbol{H}$ in the expression since, by Lemma C.5.1 for any square matrix $\boldsymbol{R}$

$$\|\boldsymbol{R} \circ \boldsymbol{H}\|_{op} \leq C\|\boldsymbol{R}\|_{op}. \tag{C.95}$$

Hence, by using (C.73), we get

$$\begin{aligned}
&\|\boldsymbol{H} \circ \boldsymbol{E}_3 \circ ((\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top)^{\circ 2}\|_{op} \leq C\|\boldsymbol{E}_3\|_{op}\|\boldsymbol{Z}_t\|_{op}^2 \\
&\|\boldsymbol{H} \circ \boldsymbol{E}_3 \circ \boldsymbol{X}_t^{\circ 2}\|_{op} \leq C\|\boldsymbol{E}_3\|_{op}\|\boldsymbol{X}_t\|_{op}^2 \\
&\|\boldsymbol{H} \circ \boldsymbol{E}_3 \circ 2(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top) \circ \boldsymbol{X}_t)\|_{op} \leq C\|\boldsymbol{E}_3\|_{op}\|\boldsymbol{X}_t\|_{op}\|\boldsymbol{Z}_t\|_{op},
\end{aligned}$$

which leads to the bound

$$\|\boldsymbol{S}_2\|_{op} \le C\|\boldsymbol{E}_3\|_{op}\left(\|\boldsymbol{X}_t\|_{op}^2 + \|\boldsymbol{Z}_t\|_{op}^2 + \|\boldsymbol{X}_t\|_{op}\|\boldsymbol{Z}_t\|_{op}\right). \tag{C.96}$$

Finally, we bound $\|\boldsymbol{S}_3\|_{op}$. Consider the term

$$\|[\boldsymbol{H} \circ \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top \circ 2(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top] \circ \boldsymbol{X}_t\|_{op}.$$

Then, by using (C.95) and (C.91), we have

$$\begin{aligned}\|\boldsymbol{H} \circ \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top\|_{op} &= \left\|\boldsymbol{H} \circ [\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top - \frac{1}{(1+\alpha)^2}\boldsymbol{I}]\right\|_{op} \\ &\le C\left\|\boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top - \frac{1}{(1+\alpha)^2}\boldsymbol{I}\right\|_{op} \le C\|\boldsymbol{Z}_t\|_{op}.\end{aligned} \tag{C.97}$$

Hence, in conjunction with (C.73), we get

$$\|\boldsymbol{H} \circ \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top \circ 2\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top\|_{op} \le C \cdot \|\boldsymbol{Z}_t\|_{op}^2,$$

which invoking (C.73) one more time gives

$$\|[\boldsymbol{H} \circ \boldsymbol{U}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{U}^\top \circ 2(\boldsymbol{U}(\boldsymbol{\Lambda}_t - \boldsymbol{I})\boldsymbol{U}^\top] \circ \boldsymbol{X}_t\|_{op} \le C\|\boldsymbol{Z}_t\|_{op}^2\|\boldsymbol{X}_t\|_{op}.$$

Furthermore, by combining (C.73) and (C.97), we get

$$\|[\boldsymbol{H} \circ \boldsymbol{\Lambda}\phi(\boldsymbol{\Lambda}_t)\boldsymbol{\Lambda}^\top] \circ \boldsymbol{X}_t^{\circ 2}\|_{op} \le C\|\boldsymbol{Z}_t\|_{op}\|\boldsymbol{X}_t\|_{op}^2.$$

Thus,

$$\|\boldsymbol{S}_3\|_{op} \le C(\|\boldsymbol{Z}_t\|_{op}^2\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op}\|\boldsymbol{X}_t\|_{op}^2). \tag{C.98}$$

Recall that, from assumptions (C.64)-(C.65), $\|\boldsymbol{X}_t\|_{op}, \|\boldsymbol{Z}_t\|_{op} \le C$. Then, by combining the bounds in (C.94), (C.96) and (C.98), the desired result readily follows. □

By exploiting the above lemmas, we are able to make the following approximation for the gradient.

Lemma C.4.3 (Gradient approximation)**.** *Assume that* (C.64) *holds, and let* $\nabla_{\boldsymbol{B}_t}$ *be given by* (C.62). *Further define* $\gamma = 1 + \alpha$ *and* $F(x) = \frac{1+x}{(\gamma+x)^2}$. *Then, for all sufficiently large* n, *with probability* $1 - 1/d^2$, *jointly for all* $t \ge 0$,

$$\left\|\frac{1}{2}\nabla_{\boldsymbol{B}_t}\boldsymbol{B}_t^\top + \alpha F(\boldsymbol{Z}_t) - \alpha\mathrm{Diag}\left(F(\boldsymbol{Z}_t)\right)(\boldsymbol{I} + \boldsymbol{Z}_t) - \frac{2\alpha}{\gamma^3}\boldsymbol{X}_t^O - \frac{\alpha}{\gamma^2}\boldsymbol{X}_t^D\right\|_{op} \le \|\boldsymbol{E}^t\|_{op}. \tag{C.99}$$

Proof of Lemma C.4.3. We start by showing that, with probability $1 - 1/d^2$, jointly for all $t \ge 0$,

$$\left\|\frac{1}{2}\nabla_{\boldsymbol{B}_t} + \alpha(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t - \alpha\mathrm{Diag}\left((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}(\boldsymbol{B}_t\boldsymbol{B}_t^\top)\right)\boldsymbol{B}_t\right\|_{op} \le \|\boldsymbol{E}^t\|_{op}. \tag{C.100}$$

Let us first consider the term $\nabla^1_{\boldsymbol{B}_t}$, which can be equivalently expressed as

$$\nabla^1_{\boldsymbol{B}_t} = 2\Big(- \boldsymbol{A}_t^\top + \mathrm{Diag}(\boldsymbol{B}_t\boldsymbol{A}_t)\boldsymbol{B}_t + \boldsymbol{T}\boldsymbol{B}_t - \mathrm{Diag}(\boldsymbol{T}(\boldsymbol{B}_t\boldsymbol{B}_t^\top))\boldsymbol{B}_t\Big),$$

where $\boldsymbol{T} = \boldsymbol{A}_t^\top\boldsymbol{A}_t - \mathrm{Diag}(\boldsymbol{A}_t^\top\boldsymbol{A}_t)$. It is then easy to verify that

$$\frac{1}{2}\nabla^1_{\boldsymbol{B}_t} = -\boldsymbol{A}_t^\top + \boldsymbol{A}_t^\top\boldsymbol{A}_t\boldsymbol{B}_t + \mathrm{Diag}(\boldsymbol{B}_t\boldsymbol{A}_t)\boldsymbol{B}_t - \mathrm{Diag}(\boldsymbol{A}_t^\top\boldsymbol{A}_t\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t. \tag{C.101}$$

Using Lemma C.4.1, we get

$$\boldsymbol{A}_t^\top\boldsymbol{A}_t = ((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1} + \boldsymbol{E}_1)\boldsymbol{B}_t\boldsymbol{B}_t^\top((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1} + \boldsymbol{E}_1), \tag{C.102}$$

where $\|\boldsymbol{E}_1\|_{op} \le \|\boldsymbol{E}^t\|_{op}$. It follows from (C.64) that $\|\boldsymbol{B}_t\boldsymbol{B}_t^\top\|_{op} \le C$. Hence, using that $\boldsymbol{B}_t\boldsymbol{B}_t^\top$ and $(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)$ commute in conjunction with $\|(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\|_{op} \le 1/\alpha$ we get

$$\boldsymbol{A}_t^\top\boldsymbol{A}_t = \boldsymbol{B}_t\boldsymbol{B}_t^\top(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} + \boldsymbol{E}_2, \tag{C.103}$$

where $\|\boldsymbol{E}_2\|_{op} \le \|\boldsymbol{E}^t\|_{op}$. Noting that $\frac{1}{\alpha+x} - \frac{\alpha}{(\alpha+x)^2} = \frac{x}{(\alpha+x)^2}$ and using the spectral theorem for the symmetric matrix $\boldsymbol{B}_t\boldsymbol{B}_t^\top$, we can further rewrite (C.103) as

$$\boldsymbol{A}_t^\top\boldsymbol{A}_t = (\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1} - \alpha(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} + \boldsymbol{E}_2. \tag{C.104}$$

With similar arguments, by Lemma C.4.1, we can write

$$\boldsymbol{B}_t\boldsymbol{A}_t = \boldsymbol{B}_t\boldsymbol{B}_t^\top(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1} + \boldsymbol{E}_3, \tag{C.105}$$

where $\|\boldsymbol{E}_3\|_{op} \le \|\boldsymbol{E}^t\|_{op}$. Noting that $1 - \frac{\alpha}{\alpha+x} = \frac{x}{\alpha+x}$, again by the spectral theorem for $\boldsymbol{B}_t\boldsymbol{B}_t^\top$, we get

$$\boldsymbol{B}_t\boldsymbol{A}_t = \boldsymbol{I} - \alpha(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1} + \boldsymbol{E}_3, \tag{C.106}$$

and, consequently, we obtain

$$\mathrm{Diag}(\boldsymbol{B}_t\boldsymbol{A}_t)\boldsymbol{B}_t = \boldsymbol{B}_t - \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1})\boldsymbol{B}_t + \boldsymbol{E}_4, \tag{C.107}$$

where $\|\boldsymbol{E}_4\|_{op} \le \|\boldsymbol{E}^t\|_{op}$. Using (C.104) and $1 - \frac{\alpha}{\alpha+x} = \frac{1}{x+\alpha}$, we get

$$\begin{aligned}\mathrm{Diag}(\boldsymbol{A}_t^\top\boldsymbol{A}_t\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t &= \mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t \\ &\qquad - \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t + \boldsymbol{E}_5 \\ &= \boldsymbol{B}_t - \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1})\boldsymbol{B}_t \\ &\qquad - \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t + \boldsymbol{E}_5,\end{aligned} \tag{C.108}$$

where $\|\boldsymbol{E}_5\|_{op} \le \|\boldsymbol{E}^t\|_{op}$.

With this in mind, we get back to (C.101). Combining the results of (C.104), (C.107) and (C.108) we get

$$\begin{aligned}\nabla^1_{\boldsymbol{B}_t} &= \underbrace{-(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\boldsymbol{B}_t}_{-\boldsymbol{A}_t^\top} + \underbrace{(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1}\boldsymbol{B}_t - \alpha(\alpha + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t}_{\boldsymbol{A}_t^\top\boldsymbol{A}_t\boldsymbol{B}_t} \\ &\quad + \underbrace{\boldsymbol{B}_t - \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1})\boldsymbol{B}_t}_{\mathrm{Diag}(\boldsymbol{B}_t\boldsymbol{A}_t)\boldsymbol{B}_t} \\ &\quad \underbrace{-\boldsymbol{B}_t + \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-1})\boldsymbol{B}_t + \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t}_{-\mathrm{Diag}(\boldsymbol{A}_t^\top\boldsymbol{A}_t\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t} + \boldsymbol{E}_6 \\ &= -\alpha(\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t + \alpha\mathrm{Diag}((\alpha\boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t\boldsymbol{B}_t^\top)\boldsymbol{B}_t + \boldsymbol{E}_6,\end{aligned} \tag{C.109}$$

where $\|\boldsymbol{E}_6\|_{op} \leq \|\boldsymbol{E}^t\|_{op}$.

Let us now analyze the second part of the gradient which involves terms of the form below for $\ell \geq 3$:

$$\nabla^{2,k,\ell}_{\boldsymbol{B}_t} := c_\ell^2 \cdot \ell \cdot \sum_{j \neq k} \langle \boldsymbol{a}_k, \boldsymbol{a}_j \rangle \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle^{(\ell-1)} \boldsymbol{J}_k \boldsymbol{b}_j.$$

Now, from the fact that

$$\boldsymbol{J}_k = \boldsymbol{I} - \boldsymbol{b}_k \boldsymbol{b}_k^\top,$$

we can write

$$c_\ell^2 \cdot \ell \cdot \sum_{j \neq k} \langle \boldsymbol{a}_k, \boldsymbol{a}_j \rangle \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle^{(\ell-1)} \boldsymbol{J}_k \boldsymbol{b}_j = c_\ell^2 \cdot \ell \cdot \sum_{j \neq k} \langle \boldsymbol{a}_k, \boldsymbol{a}_j \rangle \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle^{(\ell-1)} (\boldsymbol{b}_j - \langle \boldsymbol{b}_k, \boldsymbol{b}_j \rangle \boldsymbol{b}_k). \tag{C.110}$$

The second term of the RHS gives the following contribution to the $\boldsymbol{B}_t$ update

$$\mathsf{Diag}(\boldsymbol{A}_t^\top \boldsymbol{A}_t (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}) \boldsymbol{B}_t.$$

By recalling that $\|\boldsymbol{A}_t^\top \boldsymbol{A}_t\|_{op} \leq C$ and $\|\boldsymbol{B}_t\|_{op} \leq C$, we have

$$\begin{aligned} \|\mathsf{Diag}(\boldsymbol{A}_t^\top \boldsymbol{A}_t (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}) \boldsymbol{B}_t\|_{op} &\leq C \|\boldsymbol{A}_t^\top \boldsymbol{A}_t (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}\|_{op} \|\boldsymbol{B}_t\|_{op} \\ &\leq C \|(\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}\|_{op}. \end{aligned} \tag{C.111}$$

Now, for $\ell < 5$, we upper bound the RHS of (C.111) via Lemma C.4.1, which gives that

$$\|\mathsf{Diag}(\boldsymbol{A}_t^\top \boldsymbol{A}_t (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}) \boldsymbol{B}_t\|_{op} \leq C \|\boldsymbol{E}^t\|_{op}. \tag{C.112}$$

Furthermore, if we follow passages analogous to (C.70)-(C.71) (the only difference being that we exchange the roles of the Hadamard powers 3 and $\ell - 3$), we have that, with probability at least $1 - 1/d^2$, jointly for all $t \geq 0$ and $\ell \geq 5$,

$$\begin{aligned} \|\mathsf{Diag}(\boldsymbol{A}_t^\top \boldsymbol{A}_t (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ \ell}) \boldsymbol{B}_t\|_{op} &\leq C\sqrt{n} \|\boldsymbol{E}^t\|_{op} \left(\frac{\mathrm{poly}(\log d)}{d} \right)^{(\ell-3)/2} \\ &\leq C \|\boldsymbol{E}^t\|_{op} \left(\frac{\mathrm{poly}(\log d)}{d} \right)^{(\ell-4)/2}, \end{aligned} \tag{C.113}$$

for sufficiently large d.

Define the following quantity:

$$\boldsymbol{Y} = (\boldsymbol{A}_t^\top \boldsymbol{A}_t) \circ (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ(\ell-1)}. \tag{C.114}$$

In this view, the first term in (C.110) can be written as $\boldsymbol{Y}\boldsymbol{B}_t$. For $l < 5$, by Lemma C.4.2 we have that $\|\boldsymbol{Y}\|_{op} \leq \|\boldsymbol{E}^t\|_{op}$, hence $\|\boldsymbol{Y}\boldsymbol{B}_t\|_{op} \leq C\|\boldsymbol{E}^t\|_{op}$ as $\|\boldsymbol{B}_t\|_{op} \leq C$. Furthermore, with probability at least $1 - 1/d^2$, jointly for all $t \geq 0$ and $\ell \geq 5$, we have

$$\begin{aligned} \|\boldsymbol{Y}\boldsymbol{B}_t\|_{op} \leq C\|\boldsymbol{Y}\|_{op} &= C\sqrt{n} \|(\boldsymbol{A}_t^\top \boldsymbol{A}_t) \circ (\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})^{\circ 2}\|_{op} \max_{i,j} |(\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})_{i,j}|^{\ell-3} \\ &\leq \sqrt{n} \|\boldsymbol{E}^t\|_{op} \max_{i,j} |(\boldsymbol{B}_t \boldsymbol{B}_t^\top - \boldsymbol{I})_{i,j}|^{\ell-3} \\ &\leq \sqrt{n} \|\boldsymbol{E}^t\|_{op} \left[(C + C_X)^{\ell-3} \left(\frac{\mathrm{poly}(\log d)}{d} \right)^{(\ell-3)/2} \right] \\ &\leq (C + C_X)^{\ell-3} \|\boldsymbol{E}^t\|_{op} \left(\frac{\mathrm{poly}(\log d)}{d} \right)^{(\ell-4)/2}. \end{aligned} \tag{C.115}$$

Here, in the second line we use Lemma C.4.2; and in the third line we bound the off-diagonal entries of $\boldsymbol{X}_t$ via (C.64) and the off-diagonal entries of $\boldsymbol{Z}_t$ via Lemma C.5.2. Hence, by combining (C.113) and (C.115), we conclude that

$$\left\|\nabla^2_{\boldsymbol{B}_t}\right\|_{op} \le C\|\boldsymbol{E}^t\|_{op} + \|\boldsymbol{E}^t\|_{op}\sum_{\ell=5}^{\infty}(C+C_X)^{\ell-3}c_\ell^2\,\ell\left(\frac{\text{poly}(\log d)}{\sqrt{d}}\right)^{\ell-4} \le C\|\boldsymbol{E}^t\|_{op}, \quad \text{(C.116)}$$

where we used that the series $\sum_{\ell=5}^{\infty}(C+C_X)^{\ell-3}c_\ell^2\,\ell\left(\frac{(\text{poly}(\log d)}{\sqrt{d}}\right)^{\ell-4}$ converges to a finite value for all sufficiently large d, since $(C+C_X)\frac{\text{poly}(\log d)}{\sqrt{d}} < 1$. This finishes the proof of (C.100).

We now further analyse the gradient in (C.100). Defining $F(x) = \frac{1+x}{(\gamma+x)^2}$, with $\gamma = 1+\alpha$, we can write

$$\frac{1}{2}\nabla_{\boldsymbol{B}_t}\boldsymbol{B}_t^\top = -\alpha F(\boldsymbol{Z}_t+\boldsymbol{X}_t)+\alpha\text{Diag}\left(F(\boldsymbol{Z}_t+\boldsymbol{X}_t)\right)+\alpha\text{Diag}\left(F(\boldsymbol{Z}_t+\boldsymbol{X}_t)\right)(\boldsymbol{Z}_t+\boldsymbol{X}_t)+\boldsymbol{E}^t. \quad \text{(C.117)}$$

By a slight abuse of notation, we will denote by $F^{(l)}(0)$ the l-th derivative of the unidimensional function $F(x) = \frac{1+x}{(\gamma+x)^2}$ computed at $x=0$. Here, $F(\boldsymbol{Z}_t+\boldsymbol{X}_t)$ is defined by the spectral theorem (note that indeed $\boldsymbol{Z}_t+\boldsymbol{X}_t = \boldsymbol{B}_t\boldsymbol{B}_t^\top - \boldsymbol{I}$ is symmetric).

We will now compute the error we incur if in (C.117) we replace $F(\boldsymbol{X}_t+\boldsymbol{Z}_t)$ by $F(\boldsymbol{Z}_t)$. We first consider the case when $\|\boldsymbol{Z}_t\|_{op} > \frac{\gamma}{3}$. In this case, we have that

$$\left\|F(\boldsymbol{Z}_t+\boldsymbol{X}_t) - F(\boldsymbol{Z}_t) - F^{(1)}(0)\boldsymbol{X}_t\right\|_{op} \le C\left\|\boldsymbol{X}_t\right\|_{op} \le C\left\|\boldsymbol{Z}_t\right\|_{op}\left\|\boldsymbol{X}_t\right\|_{op}. \quad \text{(C.118)}$$

Here, the second inequality trivially holds since $\|\boldsymbol{Z}_t\|_{op} > \frac{\gamma}{3}$. To prove the first inequality, let DF be the derivative of the matrix-valued function $F(\boldsymbol{M}) = (\boldsymbol{I}+\boldsymbol{M})(\gamma\boldsymbol{I}+\boldsymbol{M})^{-2}$. Then, by evaluating this derivative for $\boldsymbol{M} = \boldsymbol{Z}_t$ in the direction of $\boldsymbol{X}_t$, we obtain

$$\begin{aligned} DF(\boldsymbol{Z}_t)\,\boldsymbol{X}_t = &-(\boldsymbol{I}+\boldsymbol{Z}_t)(\gamma\boldsymbol{I}+\boldsymbol{Z}_t)^{-1}\boldsymbol{X}_t(\gamma\boldsymbol{I}+\boldsymbol{Z}_t)^{-2} \\ &-(\boldsymbol{I}+\boldsymbol{Z}_t)(\gamma\boldsymbol{I}+\boldsymbol{Z}_t)^{-2}\boldsymbol{X}_t(\gamma\boldsymbol{I}+\boldsymbol{Z}_t)^{-1} + \boldsymbol{X}_t(\gamma\boldsymbol{I}+\boldsymbol{Z}_t)^{-2}. \end{aligned} \quad \text{(C.119)}$$

To verify this expression we first note that the derivative of the function $G(\boldsymbol{M}) = \boldsymbol{M}^{-1}$ in the direction of $\boldsymbol{X}$ is given by $DG(\boldsymbol{M})\boldsymbol{X} = -\boldsymbol{M}^{-1}\boldsymbol{X}\boldsymbol{M}^{-1}$. Now, (C.119) easily follows from the product rule applied to $F(\boldsymbol{Z}) = (\boldsymbol{I}+\boldsymbol{Z})(\gamma\boldsymbol{I}+\boldsymbol{Z})^{-1}(\gamma\boldsymbol{I}+\boldsymbol{Z})^{-1}$. By the assumptions in (C.64), we have that $\boldsymbol{Z}_t, (\gamma\boldsymbol{I}+\boldsymbol{Z}_t)^{-1}$ are uniformly bounded, hence the map DF is uniformly bounded as well. This implies that

$$\left\|F(\boldsymbol{Z}_t+\boldsymbol{X}_t) - F(\boldsymbol{Z}_t)\right\|_{op} \le C\left\|\boldsymbol{X}_t\right\|_{op}.$$

As $\left\|F^{(1)}(0)\boldsymbol{X}_t\right\|_{op} \le C\left\|\boldsymbol{X}_t\right\|_{op}$, we readily obtain (C.118).

Now we consider the case where $\|\boldsymbol{Z}_t\|_{op} \le \frac{\gamma}{3}$. First note that, by (C.64), $\|\boldsymbol{X}_t\|_{op} \le \frac{\gamma}{3}$. Hence,

$$F(\boldsymbol{Z}_t+\boldsymbol{X}_t) = \sum_{\ell=0}^{\infty} F^{(\ell)}(0)\frac{(\boldsymbol{Z}_t+\boldsymbol{X}_t)^\ell}{\ell!}.$$

The series above converges absolutely since $F^{(\ell)}(0)$ scales as $\frac{\ell!}{\gamma^\ell}\text{poly}(\ell)$. To see this, first we note that, if $h(x) = \frac{1}{(\gamma+x)^2}$, then $h^{(\ell)}(0) = (-1)^\ell(\ell+1)!\frac{1}{\gamma^{\ell+2}}$. Thus, by the product rule,

$F^{(\ell)}(0) = (-1)^\ell(\ell+1)!\frac{1}{\gamma^{\ell+2}} + (-1)^{\ell-1}\ell!\frac{1}{\gamma^{\ell+1}}$ which has the desired asymptotic behaviour. Expanding the brackets and applying the triangle inequality yields

$$\begin{aligned}\left\|F(\boldsymbol{Z}_t+\boldsymbol{X}_t) - \sum_{\ell=0}^{\infty} F^{(\ell)}(0)\frac{\boldsymbol{Z}_t^\ell}{\ell!} - F^{(1)}(0)\boldsymbol{X}_t\right\|_{op} &\leq \sum_{\ell=2}^{\infty} F^{(\ell)}(0)\frac{\|\boldsymbol{X}_t\|_{op}^\ell}{\ell!}\\ &+\sum_{\ell=2}^{\infty} F^{(\ell)}(0)\frac{1}{\ell!}\sum_{i=1}^{\ell-1}\binom{\ell}{i}\|\boldsymbol{Z}_t\|_{op}^i\|\boldsymbol{X}_t\|_{op}^{\ell-i}.\end{aligned}$$

As $\|\boldsymbol{Z}_t\|_{op}, \|\boldsymbol{X}_t\|_{op} \leq \frac{\gamma}{3}$, we have

$$\sum_{\ell=2}^{\infty} F^{(\ell)}(0)\frac{\|\boldsymbol{X}_t\|_{op}^\ell}{\ell!} \leq \|\boldsymbol{X}_t\|_{op}^2\sum_{\ell=2}^{\infty} F^{(\ell)}(0)\left(\frac{\gamma}{3}\right)^{\ell-2}\frac{1}{\ell!} \leq C\|\boldsymbol{X}_t\|_{op}^2,$$

and

$$\begin{aligned}\sum_{\ell=2}^{\infty} F^{(\ell)}(0)\frac{1}{\ell!}\sum_{i=1}^{\ell-1}\binom{\ell}{i}\|\boldsymbol{Z}_t\|_{op}^i\|\boldsymbol{X}_t\|_{op}^{\ell-i} &\leq \sum_{\ell=2}^{\infty} F^{(\ell)}(0)\frac{1}{\ell!}2^\ell\left(\frac{\gamma}{3}\right)^{\ell-2}\|\boldsymbol{Z}_t\|_{op}\|\boldsymbol{X}_t\|_{op}\\ &\leq C\|\boldsymbol{Z}_t\|_{op}\|\boldsymbol{X}_t\|_{op}.\end{aligned}$$

By combining the last three expressions and using that

$$F(\boldsymbol{Z}_t) = \sum_{\ell=0}^{\infty} F^{(\ell)}(0)\frac{\boldsymbol{Z}_t^\ell}{\ell!},$$

we obtain

$$\left\|F(\boldsymbol{X}_t+\boldsymbol{Z}_t) - F(\boldsymbol{Z}_t) - F^{(1)}(0)\boldsymbol{X}_t\right\|_{op} \leq C\left(\|\boldsymbol{X}_t\|_{op}\|\boldsymbol{Z}_t\|_{op} + \|\boldsymbol{X}_t\|_{op}^2\right). \qquad \text{(C.120)}$$

As the map DF is uniformly bounded, we have

$$\|F(\boldsymbol{Z}_t) - F(0)\boldsymbol{I}\|_{op} \leq C\|\boldsymbol{Z}_t\|_{op}. \qquad \text{(C.121)}$$

By combining (C.120), (C.121) and (C.117), we obtain

$$\begin{aligned}\frac{1}{2}\nabla_{\boldsymbol{B}_t}\boldsymbol{B}_t^\top &= -\alpha F(\boldsymbol{Z}_t) + \alpha\mathrm{Diag}\left(F(\boldsymbol{Z}_t)\right)(\boldsymbol{I}+\boldsymbol{Z}_t) - \alpha F^{(1)}(0)\boldsymbol{X}_t\\ &\quad + \alpha\mathrm{Diag}\left(\boldsymbol{X}_t F^{(1)}(0)\right) + \alpha\boldsymbol{X}_t F(0) + \boldsymbol{E}^t.\end{aligned} \qquad \text{(C.122)}$$

Using that $F(0) = \frac{1}{\gamma^2}$ and $F^{(1)}(0) = \frac{1}{\gamma^2}(1-\frac{2}{\gamma})$, we finally obtain

$$\frac{1}{2}\nabla_{\boldsymbol{B}_t}\boldsymbol{B}_t^\top = -\alpha F(\boldsymbol{Z}_t) + \alpha\mathrm{Diag}\left(F(\boldsymbol{Z}_t)\right)(\boldsymbol{I}+\boldsymbol{Z}_t) + \frac{2\alpha}{\gamma^3}\boldsymbol{X}_t^O + \frac{\alpha}{\gamma^2}\boldsymbol{X}_t^D + \boldsymbol{E}^t, \qquad \text{(C.123)}$$

which concludes the proof. □

Now let us return to the update equation of $\boldsymbol{B}_t\boldsymbol{B}_t^\top$ during the gradient step

$$\begin{aligned}\boldsymbol{B}_t'\boldsymbol{B}_t'^\top &= (\boldsymbol{B}_t - \eta\nabla_{\boldsymbol{B}_t})(\boldsymbol{B}_t - \eta\nabla_{\boldsymbol{B}_t})^\top\\ &= \boldsymbol{B}_t\boldsymbol{B}_t^\top - \eta\cdot\nabla_{\boldsymbol{B}_t}\boldsymbol{B}_t^\top - \eta\cdot\boldsymbol{B}_t(\nabla_{\boldsymbol{B}_t})^\top + \eta^2\cdot\nabla_{\boldsymbol{B}_t}(\nabla_{\boldsymbol{B}_t})^\top.\end{aligned} \qquad \text{(C.124)}$$

Note that we can control the terms $\boldsymbol{B}_t(\nabla_{\boldsymbol{B}_t})^\top$ and $\nabla_{\boldsymbol{B}_t}\boldsymbol{B}_t^\top$ via Lemma C.4.3. In this view, it remains to argue that the contribution of the term $\eta^2\cdot\nabla_{\boldsymbol{B}_t}(\nabla_{\boldsymbol{B}_t})^\top$ and of the projection step are of order $\eta\|\boldsymbol{E}^t\|_{op}$. For convenience of the upcoming lemmas we define the following quantity:

$$\widetilde{\nabla}_{\boldsymbol{B}_t} := 2\left(-\alpha(\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\boldsymbol{B}_t + \alpha\mathrm{Diag}\left((\alpha\boldsymbol{I}+\boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}(\boldsymbol{B}_t\boldsymbol{B}_t^\top)\right)\boldsymbol{B}_t\right). \qquad \text{(C.125)}$$

Lemma C.4.4. *Assume that* (C.64) *holds, and let* $\nabla_{\boldsymbol{B}_t}$ *be given by* (C.62) *with* $\eta \le C/\sqrt{d}$. *Then, for all sufficiently large* n, *with probability* $1 - 1/d^2$, *jointly for all* $t \ge 0$*:*

$$\eta^2 \left\| \nabla_{\boldsymbol{B}_t} (\nabla_{\boldsymbol{B}_t})^\top \right\|_{op} \le \eta \left\| \boldsymbol{E}^t \right\|_{op}.$$

Proof of Lemma C.4.4. We start by showing that

$$\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op} \le C(\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op}). \tag{C.126}$$

Recall that $\|\boldsymbol{B}_t\|_{op}, \|(\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2}\|_{op} \le C$. Hence, the following chain of inequalities holds

$$\begin{aligned}
\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op} &\le \|\boldsymbol{B}_t\|_{op} \cdot \left\| -\alpha(\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} + \alpha \mathrm{Diag}\left((\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} (\boldsymbol{B}_t\boldsymbol{B}_t^\top) \right) \right\|_{op} \\
&\le C \cdot \Big\| - \alpha(\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} (\boldsymbol{I} - \boldsymbol{B}_t\boldsymbol{B}_t^\top + \boldsymbol{B}_t\boldsymbol{B}_t^\top) \\
&\qquad + \alpha \mathrm{Diag}\left((\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} (\boldsymbol{B}_t\boldsymbol{B}_t^\top) \right) \Big\|_{op} \\
&\le C \Big(\left\| (\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} (\boldsymbol{Z}_t + \boldsymbol{X}_t) \right\|_{op} \\
&\qquad + \left\| (\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} \boldsymbol{B}_t\boldsymbol{B}_t^\top - \mathrm{Diag}\left((\alpha \boldsymbol{I} + \boldsymbol{B}_t\boldsymbol{B}_t^\top)^{-2} (\boldsymbol{B}_t\boldsymbol{B}_t^\top) \right) \right\|_{op} \Big) \\
&\le C \left(\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op} + \|F(\boldsymbol{X}_t + \boldsymbol{Z}_t) - \mathrm{Diag}(F(\boldsymbol{X}_t + \boldsymbol{Z}_t))\|_{op} \right),
\end{aligned} \tag{C.127}$$

where we recall the definition $F(x) = \frac{1+x}{(\gamma+x)^2}$, with $\gamma = 1 + \alpha$. By combining (C.120) and (C.121) (in the proof of Lemma C.4.3), we have

$$\|F(\boldsymbol{X}_t + \boldsymbol{Z}_t) - F(0)\boldsymbol{I}\|_{op} \le C(\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op}),$$

As $\|\mathrm{Diag}(\boldsymbol{M})\|_{op} \le C\|\boldsymbol{M}\|_{op}$ for any matrix $\boldsymbol{M}$, we also have that

$$\|\mathrm{Diag}(F(\boldsymbol{X}_t + \boldsymbol{Z}_t)) - F(0)\boldsymbol{I}\|_{op} \le C(\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op}).$$

Hence,

$$\|F(\boldsymbol{X}_t + \boldsymbol{Z}_t) - \mathrm{Diag}(F(\boldsymbol{X}_t + \boldsymbol{Z}_t))\|_{op} \le C(\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op}),$$

which finishes the proof of (C.126).

At this point, recall from (C.100) and (C.125) that

$$\left\| \nabla_{\boldsymbol{B}_t} - \widetilde{\nabla}_{\boldsymbol{B}_t} \right\|_{op} \le \left\| \boldsymbol{E}^t \right\|_{op}. \tag{C.128}$$

Thus,

$$\left\| \nabla_{\boldsymbol{B}_t} \nabla_{\boldsymbol{B}_t}^\top \right\|_{op} \le 2 \left\| \widetilde{\nabla}_{\boldsymbol{B}_t} \boldsymbol{E}^t \right\|_{op} + \left\| \widetilde{\nabla}_{\boldsymbol{B}_t} (\widetilde{\nabla}_{\boldsymbol{B}_t})^\top \right\|_{op} + \left\| (\boldsymbol{E}^t)^2 \right\|_{op}.$$

Recalling the previous bound on $\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op}$ in (C.126) and using the assumptions in (C.64), we get that

$$\left\| \widetilde{\nabla}_{\boldsymbol{B}_t} \boldsymbol{E}^t \right\|_{op}, \ \left\| \boldsymbol{E}^t \right\|_{op}^2 \le C\|\boldsymbol{E}^t\|_{op},$$

and

$$\begin{aligned}
\eta^2 \|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op}^2 &\le C\eta (\|\boldsymbol{X}_t\|_{op}^2 + \|\boldsymbol{X}_t\|_{op} \|\boldsymbol{Z}_t\|_{op}) + C\eta^2 \|\boldsymbol{Z}_t\|_{op}^2 \\
&\le C\eta \left(\frac{1}{\sqrt{d}} \|\boldsymbol{Z}_t\|_{op} + \|\boldsymbol{X}_t\|_{op}^2 + \|\boldsymbol{X}_t\|_{op} \|\boldsymbol{Z}_t\|_{op} \right) \le C\eta \left\| \boldsymbol{E}^t \right\|_{op},
\end{aligned} \tag{C.129}$$

where we have also used that $\eta \le C/\sqrt{d}$. This concludes the proof. □

The next lemma controls the contribution of the projection step.

Lemma C.4.5 (Projection step). *Assume that* (C.64) *holds and* $\eta \le C/\sqrt{d}$*. Then, for all sufficiently large* n*, with probability* $1-1/d^2$*, jointly for all* $t \ge 0$*:*

$$\|\mathrm{proj}(\boldsymbol{B}_t') - \boldsymbol{B}_t'\|_{op} \le \eta \left\|\boldsymbol{E}^t\right\|_{op},$$

which implies that, by differentiability of the bilinear form,

$$\|\mathrm{proj}(\boldsymbol{B}_t')\mathrm{proj}(\boldsymbol{B}_t')^\top - \boldsymbol{B}_t'(\boldsymbol{B}_t')^\top\|_{op} \le \eta \left\|\boldsymbol{E}^t\right\|_{op}.$$

Proof of Lemma C.4.5. Recall that the objective (C.60) does not depend on the norm of $\{\boldsymbol{b}_i\}_{i=1}^n$, hence $(\nabla_{\boldsymbol{B}_t})_{i,:}$ is orthogonal to $(\boldsymbol{B}_t)_{i,:}$, which implies that

$$\mathrm{proj}_i(\boldsymbol{B}_t') = \frac{(\boldsymbol{B}_t)_{i,:} - \eta(\nabla_{\boldsymbol{B}_t})_{i,:}}{\sqrt{1+\eta^2\|(\nabla_{\boldsymbol{B}_t})_{i,:}\|^2}}.$$

Let us define

$$\boldsymbol{D}_t := \mathrm{Diag}\left(\frac{1}{\sqrt{1+\eta^2\|(\nabla_{\boldsymbol{B}_t})_{1,:}\|^2}}, \dots, \frac{1}{\sqrt{1+\eta^2\|(\nabla_{\boldsymbol{B}_t})_{n,:}\|^2}}\right).$$

Then, we obtain the following compact form:

$$\mathrm{proj}(\boldsymbol{B}_t') = \boldsymbol{D}_t(\boldsymbol{B}_t - \eta\nabla_{\boldsymbol{B}_t}) = \boldsymbol{D}_t\boldsymbol{B}_t'.$$

In this view, it remains to bound $\|\boldsymbol{D}_t - \boldsymbol{I}\|_{op}$. In more details, by (C.126) and (C.128), we have

$$\|\nabla_{\boldsymbol{B}_t}\|_{op} \le \|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op} + \|\boldsymbol{E}^t\|_{op} \le C(\|\boldsymbol{X}_t\|_{op} + \|\boldsymbol{Z}_t\|_{op} + \|\boldsymbol{E}^t\|_{op}) \le C',$$

where $C' > 0$ is a universal constant (independent of C_X, n, d). Hence, by recalling that $\|\boldsymbol{B}_t\|_{op} \le C$ by assumption (C.64), we have

$$\|\mathrm{proj}(\boldsymbol{B}_t') - \boldsymbol{B}_t'\|_{op} = \|(\boldsymbol{D}_t - \boldsymbol{I})(\boldsymbol{B}_t - \eta\nabla_{\boldsymbol{B}_t})\|_{op} \le C\left\|\boldsymbol{D}_t - \boldsymbol{I}\right\|_{op}.$$

Note that function $1/\sqrt{1+x}$ is differentiable at 0, hence, we have that for small enough η (which follows from $\eta \le C/\sqrt{d}$):

$$\left|\frac{1}{\sqrt{1+\eta^2\|(\nabla_{\boldsymbol{B}_t})_{i,:}\|^2}} - 1\right| \le C\eta^2\|(\nabla_{\boldsymbol{B}_t})_{i,:}\|^2.$$

In this view, we have

$$\|\boldsymbol{D}_t - \boldsymbol{I}\|_{op} \le C\eta^2\|\nabla_{\boldsymbol{B}_t}\|_{op}^2 \le C\eta^2\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op}^2 + C\eta^2\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|\|\boldsymbol{E}^t\|_{op} + C\eta^2\|\boldsymbol{E}^t\|^2.$$

Inspecting each term one by one and applying (C.126) in conjunction with $\eta \le C/\sqrt{d}$ gives that

$$\begin{aligned}
&\eta^2\left\|\boldsymbol{E}^t\right\|_{op}^2 \le C\eta\left\|\boldsymbol{E}^t\right\|_{op},\\
&\eta^2\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|\|\boldsymbol{E}^t\|_{op} \le C\eta\|\boldsymbol{E}^t\|_{op},\\
&\eta^2\|\widetilde{\nabla}_{\boldsymbol{B}_t}\|_{op}^2 \le C\eta\left\|\boldsymbol{E}^t\right\|_{op},
\end{aligned}$$

where in the last step we have used (C.129). This concludes the proof. □

In this view, using (C.124) and Lemmas C.4.3, C.4.4 and C.4.5, we obtain

$$\begin{aligned}\boldsymbol{I}+\boldsymbol{Z}_{t+1}+\boldsymbol{X}_{t+1}=\boldsymbol{B}_{t+1}\boldsymbol{B}_{t+1}^\top=\boldsymbol{I}+\boldsymbol{Z}_t+\boldsymbol{X}_t+4\eta\alpha F(\boldsymbol{Z}_t)-2\eta\alpha\mathrm{Diag}(F(\boldsymbol{Z}_t))(\boldsymbol{I}+\boldsymbol{Z}_t)\\ -2\eta\alpha(\boldsymbol{I}+\boldsymbol{Z}_t)\mathrm{Diag}(F(\boldsymbol{Z}_t))-\frac{8\alpha\eta}{\gamma^3}\boldsymbol{X}_t^O-\frac{4\alpha\eta}{\gamma^2}\boldsymbol{X}_t^D+\eta\boldsymbol{E}^t.\end{aligned}\tag{C.130}$$

Furthermore, we have that

$$\begin{aligned}\mathrm{Diag}(F(\boldsymbol{Z}_t))(\boldsymbol{I}+\boldsymbol{Z}_t)&=\left(\mathrm{Diag}(F(\boldsymbol{Z}_t)-F(0)\boldsymbol{I})+F(0)\boldsymbol{I}\right)(\boldsymbol{I}+\boldsymbol{Z}_t)\\&=\frac{1}{\gamma^2}(\boldsymbol{I}+\boldsymbol{Z}_t)+\left(\mathrm{Diag}(F(\boldsymbol{Z}_t)-F(0)\boldsymbol{I})\right)(\boldsymbol{I}+\boldsymbol{Z}_t)\\&=\frac{1}{\gamma^2}(\boldsymbol{I}+\boldsymbol{Z}_t)+\left(\frac{1}{n}\mathrm{Tr}\left[F(\boldsymbol{Z}_t)-F(0)\boldsymbol{I}\right]+\boldsymbol{D}_t'\right)(\boldsymbol{I}+\boldsymbol{Z}_t),\end{aligned}\tag{C.131}$$

where $\boldsymbol{D}_t'$ is a diagonal matrix such that, with probability at least $1-1/d^2$, its entries are upper bounded in modulus by $\frac{C\log d}{\sqrt{d}}\|\boldsymbol{Z}_t\|_{op}^{1/2}$. The last passage follows from Lemma C.5.2. Note that $\frac{1}{\gamma^2}(\boldsymbol{I}+\boldsymbol{Z}_t)=\frac{1}{n}\mathrm{Tr}\left[F(0)\boldsymbol{I}\right]$ and recall that $\|\boldsymbol{Z}_t\|_{op}\leq C$. Hence, (C.131) implies that

$$\mathrm{Diag}(F(\boldsymbol{Z}_t))(\boldsymbol{I}+\boldsymbol{Z}_t)=\frac{1}{n}\mathrm{Tr}\left[F(\boldsymbol{Z}_t)\right](\boldsymbol{I}+\boldsymbol{Z}_t)+\boldsymbol{E}^t.\tag{C.132}$$

Similarly, we have that

$$(\boldsymbol{I}+\boldsymbol{Z}_t)\mathrm{Diag}(F(\boldsymbol{Z}_t))=\frac{1}{n}\mathrm{Tr}\left[F(\boldsymbol{Z}_t)\right](\boldsymbol{I}+\boldsymbol{Z}_t)+\boldsymbol{E}^t.\tag{C.133}$$

By combining (C.132)-(C.133) with (C.130) and using that $\boldsymbol{X}_t=\boldsymbol{X}_t^O+\boldsymbol{X}_t^D$, we get

$$\begin{aligned}\boldsymbol{Z}_{t+1}+\boldsymbol{X}_{t+1}=\left(1-\frac{8\alpha}{\gamma^3}\eta\right)\boldsymbol{X}_t^O+\left(1-\frac{4\alpha}{\gamma^2}\eta\right)\boldsymbol{X}_t^D+\boldsymbol{Z}_t+4\eta\alpha F(\boldsymbol{Z}_t)\\-4\eta\alpha\frac{1}{n}\mathrm{Tr}\left[F(\boldsymbol{Z}_t)\right](\boldsymbol{I}+\boldsymbol{Z}_t)+\eta\boldsymbol{E}^t.\end{aligned}\tag{C.134}$$

Hence, we can write the following system capturing the dynamics of the spectrum $\boldsymbol{Z}_t$ and of the errors $(\boldsymbol{X}_t^O,\boldsymbol{X}_t^D)$

$$\boldsymbol{Z}_{t+1}=\boldsymbol{Z}_t+4\eta\alpha F(\boldsymbol{Z}_t)-4\eta\alpha\frac{1}{n}\mathrm{Tr}\left[F(\boldsymbol{Z}_t)\right](\boldsymbol{I}+\boldsymbol{Z}_t),\tag{C.135}$$

$$\boldsymbol{X}_{t+1}^D=\left(1-\frac{4\alpha}{\gamma^2}\eta\right)\boldsymbol{X}_t^D+\eta\boldsymbol{E}^t,\tag{C.136}$$

$$\boldsymbol{X}_{t+1}^O=\left(1-\frac{8\alpha}{\gamma^3}\eta\right)\boldsymbol{X}_t^O+\eta\boldsymbol{E}^t.\tag{C.137}$$

Here, the operator norm of $\boldsymbol{E}^t$ is upper bounded as in (C.66), where we recall that the constant C is uniformly bounded in t.

In the view of (C.135), one can readily see that the updates on the spectrum of $\boldsymbol{Z}_t$ follow the one described in Lemma C.5.3 and, thus, converges exponentially. This means that the set of assumptions on $\boldsymbol{Z}_t$ in (C.64) is satisfied by suitably picking C.

Now it only remains to take care of $\boldsymbol{X}_t$. If we write $x_t^D = \left\|\boldsymbol{X}_t^D\right\|_{op}, x_t^O = \left\|\boldsymbol{X}_t^O\right\|_{op}, z_t = \|\boldsymbol{Z}_t\|_{op}^{1/2}$, then recalling the definition of $\boldsymbol{E}_t$ in (C.66), (C.136), (C.137) we have that

$$x_{t+1}^D \leq \left(1 - \frac{4\alpha}{\gamma^2}\eta\right) x_t^D + \eta C_D \left(\frac{\operatorname{poly}(\log d)}{\sqrt{d}} \cdot z_t + (x_t^D + x_t^O)^2 + (x_t^D + x_t^O) z_t\right) \tag{C.138}$$

$$x_{t+1}^O \leq \left(1 - \frac{8\alpha}{\gamma^3}\eta\right) x_t^O + \eta C_O \left(\frac{\operatorname{poly}(\log d)}{\sqrt{d}} \cdot z_t + (x_t^D + x_t^O)^2 + (x_t^D + x_t^O) z_t\right). \tag{C.139}$$

Since both of these recursive bounds are monotone in x_t^D, x_t^O, we can dominate them as follows. If we recursively define x_t by

$$\begin{aligned} x_{t+1} = &\left(1 - \eta \min\left\{\frac{4\alpha}{\gamma^2}, \frac{8\alpha}{\gamma^3}\right\}\right) x_t \\ &+ \eta \max\{C_D, C_O\} \left(\frac{\operatorname{poly}(\log d)}{\sqrt{d}} \cdot z_t + (x_t + x_t)^2 + (x_t + x_t) z_t\right), \end{aligned} \tag{C.140}$$

then by monotonicity $\max\{x_t^D, x_t^O\} \leq x_t$. Thus, we only need to analyse the recursion (C.140), which we do in the following lemma. Note that the condition $z_t \leq Ce^{-ct\eta}$ required by Lemma C.4.6 holds by (C.64).

Lemma C.4.6 (Error decay). *Let $\{z_t\}_{t=0}^{\infty}$ be a non-negative exponentially decaying sequence, i.e., $z_t \leq C_z e^{-\eta c_z t}$, and consider a non-negative sequence $\{x_t\}_{t=0}^{\infty}$ such that at each time-step t the following condition holds for $\eta = \Theta(1/\sqrt{d})$ and sufficiently large d:*

$$x_{t+1} = (1 - \eta c_1) x_t + \eta C_2 \cdot z_t \cdot x_t + \eta C_3 x_t^2 + \eta C_4 \cdot \frac{\operatorname{poly}(\log d)}{\sqrt{d}} \cdot z_t, \tag{C.141}$$

with $x_0 = 0$. Then, the following holds

$$x_t \leq C \frac{\operatorname{poly}(\log d)}{\sqrt{d}} \cdot T e^{-cT}, \tag{C.142}$$

where $T = t\eta$.

Proof of Lemma C.4.6. We proceed in two parts. In the first part, we show that our recursion does not blow up in $t = K/\eta$ steps. In the second part, $z_t \leq C_z \exp(-c_z K)$ will be small, which allows us to deduce (C.142).

Error does not blow up in finite time. Let $t = K/\eta$ where K is such that $K/\eta \in \mathbb{N}$. We start by analysing the simpler recursion

$$x_{t+1} = (1 - \eta c_1) x_t + \eta C_2 \cdot z_t \cdot x_t + \eta C_4 \cdot \frac{\operatorname{poly}(\log d)}{\sqrt{d}} \cdot z_t.$$

By hypothesis, $z_t \leq C_z$. Hence, we arrive to

$$x_{t+1} = (1 - \eta c_1) x_t + \eta C_2 C_z \cdot x_t + \eta C_4 C_z \frac{\operatorname{poly}(\log d)}{\sqrt{d}}.$$

Writing $C_5 = C_2 C_z - c_1$, unrolling the recursion on the RHS and using $x_0 = 0$ gives

$$\begin{aligned} x_{t+1} &= \eta C_4 C_z \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \sum_{j=0}^{t} (1+\eta C_5)^j \\ &\leq \eta C_4 C_z \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \sum_{j=0}^{K/\eta} e^{\eta C_5 j} \\ &= \eta C_4 C_z \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \cdot e^{C_5 K} \sum_{j=0}^{K/\eta} e^{-C_5 \eta (t-j)} \\ &\leq \eta C_4 C_z \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \cdot \frac{e^{C_5 K}}{1-e^{-\eta C_5}}, \end{aligned}$$

where the inequality holds for $t \leq K/\eta$ and we have used $1 + x \leq e^x$. For small enough η, we have that

$$\frac{\eta}{1-e^{-C_5\eta}} \leq \frac{2}{C_5},$$

hence, for all $t \leq K/\eta$,

$$x_{t+1} \leq 2 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \frac{C_4 C_z}{C_5} \exp(C_5 K). \tag{C.143}$$

Let us now go back to our original recursion (C.141), which contains the term x_t^2. We claim that this recursion satisfies a bound like (C.143). Assume by contradiction that it exceeds the bound

$$x_t \leq 4 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \frac{C_4 C_z}{C_5} \exp(C_5 K) \tag{C.144}$$

for the first time at step t'. Then, for all $t < t'$, (C.144) holds. Noting that $x_t^2 \leq 4 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \frac{C_4 C_z}{C_5} \exp(C_5 K) x_t$ we define $C_5' = C_2 C_z + 4 C_3 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \frac{C_4 C_z}{C_5} \exp(C_5 K) - c_1$. By unrolling the recursion exactly as before, we obtain

$$x_{t+1} \leq 2 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \frac{C_4 C_z}{C_5'} \exp(C_5' K) \leq 3 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \frac{C_4 C_z}{C_5} \exp(C_5 K), \tag{C.145}$$

for d large enough. Here, the second inequality follows for large d, since it is clear from the definitions that $|C_5 - C_5'|$ vanishes for large d. This shows that we cannot violate (C.144), thus (C.145) holds for all $t \leq K/\eta$.

Convergence of errors x_t to zero. We now choose K large enough so that

$$z_t = C_z e^{-\eta c_z t} < \frac{c_1}{2C_2}, \quad \forall t \geq K/\eta.$$

Hence, the term corresponding to $\eta C_2 z_t x_t$ can be pushed inside the $(1 - \eta c_1)x_t$ term. Consequently, we can equivalently study the following dynamics

$$x_{t+1} = (1 - \eta c_1')x_t + \eta C_3 x_t^2 + \eta C_4 C_z \frac{\mathrm{poly}(\log d)}{\sqrt{d}} e^{-\eta c_z t}, \tag{C.146}$$

where $c_1' = c_1/2$. Here, we initialize again at $t = 0$, but now starting at

$$x_0 = C_6 \frac{\mathrm{poly}(\log d)}{\sqrt{d}},$$

where $C_6 = 4\frac{C_4 C_z}{C_5}\exp(C_5 K)$, corresponding to the bound in (C.144). Rearranging we have

$$x_{t+1} = x_t + \eta\left(-c_1' x_t + C_3 x_t^2 + C_4 C_z \frac{\operatorname{poly}(\log d)}{\sqrt{d}} e^{-\eta c_z t}\right). \tag{C.147}$$

As the last term inside the brackets vanishes when $d \to \infty$, we have two roots of the polynomial inside the brackets, corresponding to the fixed points of the iteration. The left root r_l scales as

$$r_l \leq C_l \frac{\operatorname{poly}(\log d)}{\sqrt{d}} e^{-\eta c_z t},$$

and the right root r_r as

$$r_r \geq \frac{c_1'}{C_3} - C\frac{\operatorname{poly}(\log d)}{\sqrt{d}} e^{-\eta c_z t}.$$

In addition, it is easy to see that both roots are non-negative.

Next, we prove that $x_t \leq C\frac{\operatorname{poly}(\log d)}{\sqrt{d}}$ for all t. We will show this by contradiction. At initialization we have

$$x_0 = C_6 \frac{\operatorname{poly}(\log d)}{\sqrt{d}}.$$

Choose A, B as follows:

$$A := \max\{C_l, C_6\}, \quad B = C_7 A.$$

We first note that, for small enough η and large enough d, we can choose C_7 such that $x_{\tilde{t}} \leq A\frac{\operatorname{poly}(\log d)}{\sqrt{d}}$ implies $x_{\tilde{t}+1} \leq B\frac{\operatorname{poly}(\log d)}{\sqrt{d}}$. We now show that $x_t \leq B\frac{\operatorname{poly}(\log d)}{\sqrt{d}}$ for all t. To do so, assume by contradiction that $x_{t+1} > B\frac{\operatorname{poly}(\log d)}{\sqrt{d}}$. Then $x_t \in [A\frac{\operatorname{poly}(\log d)}{\sqrt{d}}, B\frac{\operatorname{poly}(\log d)}{\sqrt{d}}] \subseteq [r_l, r_r]$, thus

$$-c_1' x_t + C_3 x_t^2 + C_4 C_z \frac{\operatorname{poly}(\log d)}{\sqrt{d}} e^{-\eta c_z t} < 0.$$

Hence, from (C.147) it follows that

$$x_{t+1} \leq x_t \leq B\frac{\operatorname{poly}(\log d)}{\sqrt{d}},$$

which gives us the desired contradiction.

Thus, for all t,

$$x_t^2 \leq B\frac{\operatorname{poly}(\log d)}{\sqrt{d}} x_t.$$

This allows us to push the second term in (C.146) into the first one (for d large enough), which reduces the recursion to

$$x_{t+1} = (1 - \eta c_1'')\, x_t + \eta C_4 C_z \frac{\operatorname{poly}(\log d)}{\sqrt{d}} e^{-\eta c_z t},$$

where $c_1'' \geq c_1'/2$. By unrolling this last recursion and using $x_0 = C_6\frac{\operatorname{poly}(\log d)}{\sqrt{d}}$, we have that, for $t \geq 1$,

$$x_t = C_6 \frac{\operatorname{poly}(\log d)}{\sqrt{d}} (1 - \eta c_1'')^t + \eta C_4 C_z \frac{\operatorname{poly}(\log d)}{\sqrt{d}} \sum_{\ell=1}^{t} (1 - \eta c_1'')^{t-\ell} e^{-\eta c_z \ell} \tag{C.148}$$

$$\leq C_6 \frac{\operatorname{poly}(\log d)}{\sqrt{d}} \exp(-\eta c_1'' t) + \eta C_4 C_z \frac{\operatorname{poly}(\log d)}{\sqrt{d}} \sum_{\ell=1}^{t} e^{-\eta(c_z \ell + c_1''(t-\ell))}, \tag{C.149}$$

where the inequality follows from $1 - x \le e^{-x}$. Since the term in the exponents of the sum is a linear function in ℓ, its maximum value is attained in the endpoints. Thus,

$$x_t \le C_6 \frac{\mathrm{poly}(\log d)}{\sqrt{d}} \exp(-\eta c_1'' t) + \eta C_4 C_z \frac{\mathrm{poly}(\log d)}{\sqrt{d}} t \max\{e^{-\eta c_z t}, e^{-\eta c_1'' t}\},$$

which implies (C.142). □

By Lemma C.4.6 we know that

$$\|\boldsymbol{X}_t\|_{op} \le \frac{C}{\sqrt{d}} \cdot T e^{-cT},$$

where C is independent of C_X by definition. Hence, we can pick C_X such that, for sufficiently large d, the assumptions on $\boldsymbol{X}_t$ in (C.64) are satisfied. With this in mind, we can use Lemma C.5.3 to bound the dynamics involving $\boldsymbol{Z}_t$ and Lemma C.4.6 to claim that the error $\boldsymbol{X}_t$ vanishes at least geometrically fast. This concludes the proof of Theorem 12.

C.5 Auxiliary Results

Lemma C.5.1. *For any $\boldsymbol{R} \in \mathbb{R}^{n \times n}$ the following holds*

$$\|\boldsymbol{R} - \mathrm{diag}(\boldsymbol{R})\|_{op} \le C \|\boldsymbol{R}\|_{op}.$$

Proof. By definition of the operator norm we have that

$$\|\boldsymbol{R}\|_{op} = \sup_{\|\boldsymbol{x}\|_2 = 1} \|\boldsymbol{R}\boldsymbol{x}\|_2.$$

Note that by Cauchy-Schwarz, the following holds for $\|\boldsymbol{y}\|_2 = 1$:

$$\langle \boldsymbol{y}, \boldsymbol{R}\boldsymbol{x} \rangle \le \|\boldsymbol{R}\boldsymbol{x}\|_2,$$

and the inequality is met when $\boldsymbol{y}$ is aligned with $\boldsymbol{R}\boldsymbol{x}$. Hence, we get

$$\sup_{\|\boldsymbol{y}\|_2 = 1} \langle \boldsymbol{y}, \boldsymbol{R}\boldsymbol{x} \rangle = \|\boldsymbol{R}\boldsymbol{x}\|_2,$$

and, thus, the operator norm can be rewritten as

$$\|\boldsymbol{R}\|_{op} = \sup_{\|\boldsymbol{x}\|_2 = 1} \|\boldsymbol{R}\boldsymbol{x}\|_2 = \sup_{\|\boldsymbol{x}\|_2 = \|\boldsymbol{y}\|_2 = 1} \langle \boldsymbol{y}, \boldsymbol{R}\boldsymbol{x} \rangle.$$

Note also that $\|\mathrm{diag}(\boldsymbol{R})\|_{op}$ is equal to the maximal diagonal element (in absolute value). Hence, by letting $\boldsymbol{e}_i$ be the i-th element of the canonical basis, we get

$$\|\mathrm{diag}(\boldsymbol{R})\|_{op} = \sup_i |\boldsymbol{R}_{i,i}| \le \sup_i |\langle \boldsymbol{e}_i, \boldsymbol{R}\boldsymbol{e}_i \rangle| \le \sup_{\|\boldsymbol{x}\|_2 = \|\boldsymbol{y}\|_2 = 1} \langle \boldsymbol{y}, \boldsymbol{R}\boldsymbol{x} \rangle = \|\boldsymbol{R}\|_{op}.$$

In this view, an application of triangle inequality, i.e.,

$$\|\boldsymbol{R} - \mathrm{diag}(\boldsymbol{R})\|_{op} \le \|\boldsymbol{R}\|_{op} + \|\mathrm{diag}(\boldsymbol{R})\|_{op} \le 2 \|\boldsymbol{R}\|_{op},$$

finishes the proof. □

Lemma C.5.2. *Consider the matrix $\boldsymbol{A}_t = \boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top$, where the matrix $\boldsymbol{U}$ is distributed according to the Haar measure and it is independent from the diagonal matrix $\boldsymbol{\Lambda}_t$. Further, assume that all the diagonal entries of $\boldsymbol{\Lambda}_t$ are bounded in absolute value by a constant. Then, the following results hold.*

1. *We have that, with probability at least $1 - 1/d^2$,*

$$\max_{i\neq j} |(\boldsymbol{A}_t)_{i,j}| \le c\sqrt{\frac{\log d}{d}}, \tag{C.150}$$

for some absolute constant $c > 0$.

2. *Let $\boldsymbol{D}_t =$ diag$(\boldsymbol{A}_t)$. Then,*

$$\boldsymbol{D}_t = \alpha \boldsymbol{I} + \boldsymbol{D}'_t,$$

where

$$\alpha = \frac{1}{n}\mathrm{Tr}(\boldsymbol{\Lambda}_t),$$

and $\boldsymbol{D}'_t$ is a diagonal matrix such that, with probability at least $1 - 1/d^2$,

$$\max_{i\in[n]} |(\boldsymbol{D}'_t)_{i,i}| \le c\frac{\log d}{\sqrt{d}}. \tag{C.151}$$

3. *Assume that, for all $t \in \mathbb{N}$,*

$$\|\boldsymbol{\Lambda}_t\|_{op} \le Ce^{-c\eta t}, \tag{C.152}$$

where $c, C > 0$ are absolute constants and $\eta = \Theta(1/\sqrt{d})$. Then, with probability at least $1 - 1/d^2$,

$$\sup_{t\ge 0}\max_{i\neq j} |(\boldsymbol{A}_t)_{i,j}| \le c\sqrt{\frac{\log d}{d}}, \tag{C.153}$$

$$\sup_{t\ge 0}\max_{i\in[n]} |(\boldsymbol{D}'_t)_{i,i}| \le c\frac{\log d}{\sqrt{d}}. \tag{C.154}$$

Proof. We start by proving (C.150). Consider the metric measure space $(\mathbb{SO}(d), \|\cdot\|_F, \mathbb{P})$. Here, $\mathbb{SO}(d)$ denotes the special orthogonal group containing all $d \times d$ orthogonal matrices with determinant 1 (i.e., all rotation matrices), and $\mathbb{P}$ is the uniform probability measure on $\mathbb{SO}(d)$, i.e., the Haar measure. Given a diagonal matrix $\boldsymbol{\Lambda}_t$ and two indices $i, j \in [d]$, define $f : \mathbb{SO}(d) \to \mathbb{R}$ as

$$f(\boldsymbol{M}) = (\boldsymbol{M}\boldsymbol{\Lambda}_t\boldsymbol{M}^\top)_{i,j}. \tag{C.155}$$

Note that

$$\begin{aligned}
|f(\boldsymbol{M}) - f(\boldsymbol{M}')| &= |(\boldsymbol{M}\boldsymbol{\Lambda}_t\boldsymbol{M}^\top)_{i,j} - (\boldsymbol{M}'\boldsymbol{\Lambda}_t(\boldsymbol{M}')^\top)_{i,j}| \\
&\le |(\boldsymbol{M}\boldsymbol{\Lambda}_t\boldsymbol{M}^\top)_{i,j} - (\boldsymbol{M}'\boldsymbol{\Lambda}_t\boldsymbol{M}^\top)_{i,j}| \\
&\quad + |(\boldsymbol{M}'\boldsymbol{\Lambda}_t\boldsymbol{M}^\top)_{i,j} - (\boldsymbol{M}'\boldsymbol{\Lambda}_t(\boldsymbol{M}')^\top)_{i,j}| \\
&\le |((\boldsymbol{M} - \boldsymbol{M}')\boldsymbol{\Lambda}_t\boldsymbol{M}^\top)_{i,j}| + |(\boldsymbol{M}'\boldsymbol{\Lambda}_t(\boldsymbol{M} - \boldsymbol{M}')^\top)_{i,j}| \\
&\le \|(\boldsymbol{M} - \boldsymbol{M}')\boldsymbol{\Lambda}_t\boldsymbol{M}^\top\|_F + \|\boldsymbol{M}'\boldsymbol{\Lambda}_t(\boldsymbol{M} - \boldsymbol{M}')^\top\|_F \\
&\le 2\|\boldsymbol{M} - \boldsymbol{M}'\|_F\|\boldsymbol{\Lambda}_t\|_{op}\|\boldsymbol{M}\|_{op} \le 2\|\boldsymbol{M} - \boldsymbol{M}'\|_F\|\boldsymbol{\Lambda}_t\|_{op},
\end{aligned} \tag{C.156}$$

where in the fourth inequality we use that, for any two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, $\|\boldsymbol{A}\boldsymbol{B}\|_F \leq \|\boldsymbol{A}\|_{op}\|\boldsymbol{B}\|_F$, and in the fifth inequality we use that $\|\boldsymbol{M}\|_{op} = 1$ as $\boldsymbol{M} \in \mathbb{SO}(d)$. Hence, f has Lipschitz constant upper bounded by $2\|\boldsymbol{\Lambda}_t\|_{op}$ and an application of Theorem 5.2.7 of [Ver18] gives that

$$\mathbb{P}(|f(\boldsymbol{U}) - \mathbb{E}[f(\boldsymbol{U})]| \geq u) \leq 2\exp\left(-c_1 \frac{d\boldsymbol{u}^2}{2\|\boldsymbol{\Lambda}_t\|_{op}}\right), \tag{C.157}$$

where c_1 is a universal constant.

Let $\boldsymbol{u}_i$ denote the i-th row of $\boldsymbol{U}$. Then,

$$f(\boldsymbol{U}) = \langle \boldsymbol{u}_i, \boldsymbol{\Lambda}_t \boldsymbol{u}_j \rangle. \tag{C.158}$$

Suppose that $i \neq j$. Since $\boldsymbol{U}$ is distributed according to the Haar measure, $\boldsymbol{u}_i$ is uniform on the unit sphere and $\boldsymbol{u}_j$ is uniformly distributed on the unit sphere in the orthogonal complement of $\boldsymbol{u}_i$ (see Section 1.2 of [Mec19]). Thus, $(\boldsymbol{u}_i, \boldsymbol{u}_j)$ has the same distribution as $(-\boldsymbol{u}_i, \boldsymbol{u}_j)$, which implies that, whenever $i \neq j$

$$\mathbb{E}[f(\boldsymbol{U})] = 0. \tag{C.159}$$

By combining (C.157)-(C.159) with a union bound over i, j, we have that

$$\mathbb{P}(\max_{i \neq j} |(\boldsymbol{U}\boldsymbol{\Lambda}_t\boldsymbol{U}^\top)_{i,j}| \geq u) \leq 2d^2 \exp\left(-c_1 \frac{du^2}{2\|\boldsymbol{\Lambda}_t\|_{op}}\right). \tag{C.160}$$

As $\|\boldsymbol{\Lambda}_t\|_{op}$ is upper bounded by a universal constant, the result (C.150) readily follows.

For the second part, note that

$$(\boldsymbol{D}_t)_{i,i} = \langle \boldsymbol{u}_i, \boldsymbol{\Lambda}_t \boldsymbol{u}_i \rangle. \tag{C.161}$$

Furthermore, the following chain of equalities hold

$$\mathbb{E}[(\boldsymbol{D}_t)_{i,i}] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[(\boldsymbol{D}_t)_{i,i}] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n (\boldsymbol{D}_t)_{i,i}\right] = \frac{1}{n}\mathrm{Tr}(\boldsymbol{D}_t), \tag{C.162}$$

where the first equality uses that the $\boldsymbol{u}_i$'s have the same (marginal) distribution, and the last term does not contain an expectation since $\mathrm{Tr}(\boldsymbol{D}_t) = \mathrm{Tr}(\boldsymbol{A}_t) = \sum_{i=1}^d (\boldsymbol{\Lambda}_t)_{i,i}$, which does not depend on $\boldsymbol{U}$. Therefore, by using (C.157) and by performing a union bound over $i \in [n]$, the result (C.151) follows.

For the third part, by performing a union bound over $t \geq 0$ in (C.160), we have that (C.153) holds with probability at least

$$\begin{aligned}
2\sum_{t=0}^\infty \exp\left(-c_1 \frac{du^2}{2\|\boldsymbol{\Lambda}_t\|_{op}}\right) &\leq 2\sum_{t=0}^\infty \exp\left(-c_2\, d\, u^2\, e^{C\eta t}\right) \\
&\leq 2\sum_{t=0}^\infty \exp\left(-c_2\, d\, u^2\, e^{C\lfloor \eta t \rfloor}\right) \\
&\leq 2\left\lceil \frac{1}{\eta} \right\rceil \sum_{t=0}^\infty \exp\left(-c_2\, d\, u^2\, e^{Ct}\right) \\
&\leq C\sqrt{d}\sum_{t=0}^\infty \exp\left(-c_2\, d\, u^2\, e^{Ct}\right),
\end{aligned} \tag{C.163}$$

where the first inequality follows from (C.152) and the last one from $\eta = \Theta(1/\sqrt{d})$. Choosing $u = c\frac{\log d}{\sqrt{d}}$ we can get that $b := \exp\left(-c_2\, d\, u^2\right) < 1$ and, hence, the following holds

$$\sum_{t=0}^{\infty} \exp\left(-c_2\, d\, u^2\right)^{e^{Ct}} \leq \sum_{t=0}^{\infty} \exp\left(-c_2\, d\, u^2\right)^{Ct+1} = \frac{b}{1-b^C} \leq \frac{1}{d^3},$$

where the first inequality uses that $e^t \geq 1+t$ and the second inequality follows from the definition of b. This concludes the proof of (C.153). The proof of (C.154) uses an analogous union bound on $t \geq 0$. □

Lemma C.5.3. *Let $\lambda^0 = \{\lambda_1^0, \cdots, \lambda_n^0\}$ be a set of numbers in $\mathbb{R}$ such that*

$$\lambda_{min}^0 := \min_{i\in[n]} \lambda_i^0 \geq \delta > 0, \quad \lambda_{max}^0 := \max_{i\in[n]} \lambda_i^0 \leq M < +\infty, \quad \sum_{j=1}^{n} \lambda_j^0 = n.$$

Let the values $\{\lambda_i^t\}_{i=1}^n$ be updated according to the equation below

$$\lambda_i^{t+1} = \lambda_i^t + \eta\left(F(\lambda_i^t) - \lambda_i^t \cdot \frac{1}{n}\sum_{j=1}^{n} F(\lambda_j^t)\right) = G(\lambda_i^t, \lambda^t), \tag{C.164}$$

where $F(\cdot)$ is defined as per Lemma C.4.3, $\eta = \Theta\left(1/\sqrt{d}\right)$ and $\lambda^t := \{\lambda_1^t, \cdots, \lambda_n^t\}$. Then, for large enough d, we have

$$\left|\lambda_i^{t+1} - 1\right| \leq (1 - c\delta\cdot\eta)\left|\lambda_i^t - 1\right|$$

and thus after t iterations

$$\left|\lambda_i^t - 1\right| \leq \max\{(M-1), (1-\delta)\}\exp(-c\delta\cdot\eta t),$$

where $c, C > 0$ are constants.

Proof. We first show by induction that $\sum_{i=1}^{n} \lambda_i^t = n$ holds for all t. In fact,

$$\begin{aligned}\sum_{i=1}^{n} \lambda_i^{t+1} &= \sum_{i=1}^{n} \lambda_i + \eta\left(\sum_{i=1}^{n} F(\lambda_i^t) - \sum_{i=1}^{n} \lambda_i^t \cdot \frac{1}{n}\sum_{j=1}^{n} F(\lambda_j^t)\right) \\ &= n + \eta\left(\sum_{i=1}^{n} F(\lambda_i^t) - \sum_{j=1}^{n} F(\lambda_j^t)\right) = n.\end{aligned}$$

Now, we will show the convergence of λ_{min}^t and λ_{max}^t. To do so, we assume that $\lambda_{max}^t \leq M$ and $\lambda_{min}^t \geq \delta$ holds at time step t (we will verify this later). Define the function $g : \mathbb{R} \to \mathbb{R}$ as

$$g(x) := x + \eta\left(F(x) - x \cdot C\right). \tag{C.165}$$

By taking the derivative, we have that, for sufficiently large d,

$$g'(x) = 1 + \eta\left(F'(x) - C\right) > 0,$$

as $\|F'\|_\infty \leq C$. This implies that $g(\cdot)$ is a monotone increasing function, which gives that

$$\begin{aligned}\max_{i\in[n]} g(\lambda_i^t) &= g(\lambda_{max}^t), \\ \min_{i\in[n]} g(\lambda_i^t) &= g(\lambda_{min}^t).\end{aligned} \tag{C.166}$$

Note that the updates on λ_i^t in (C.164) have a common part for all $i \in [n]$, i.e.,

$$\left|\frac{1}{n}\sum_{j=1}^{n} F(\lambda_j^t)\right| \leq C,$$

where we used that $\|F\|_\infty \leq C$. In this view, by definition of g and (C.166), we have

$$\begin{aligned} \lambda_{max}^{t+1} &= G(\lambda_{max}^t, \lambda^t), \\ \lambda_{min}^{t+1} &= G(\lambda_{min}^t, \lambda^t), \end{aligned} \tag{C.167}$$

which means that the min/max value at the previous step are mapped to the min/max value at the next step of (C.164). Using that $\frac{1}{n}\sum\limits_{i=1}^{n}\lambda_i^t = 1$ we can write

$$\begin{aligned} \lambda_i^{t+1} &= \lambda_i^t + \eta\left(\frac{1}{n}\sum_{j=1}^{n}\lambda_j^t \cdot F(\lambda_i^t) - \lambda_i^t \cdot \frac{1}{n}\sum_{j=1}^{n}F(\lambda_j^t)\right) \\ &= \lambda_i^t + \eta\left(\frac{1}{n}\sum_{j=1}^{n}\left[\frac{\lambda_j^t\lambda_i^t}{(\alpha+\lambda_i^t)^2} - \frac{\lambda_i^t\lambda_j^t}{(\alpha+\lambda_j^t)^2}\right]\right) \\ &= \lambda_i^t + \eta\left(\frac{1}{n}\sum_{j=1}^{n}\lambda_i^t\lambda_j^t\left(\frac{(2\alpha+\lambda_i^t+\lambda_j^t)(\lambda_j^t-\lambda_i^t)}{(\alpha+\lambda_i^t)^2(\alpha+\lambda_j^t)^2}\right)\right). \end{aligned} \tag{C.168}$$

Recall that we assumed $\lambda_{max}^t \leq M$ and $\lambda_{min}^t \geq \delta$. In this view, we get the following bound

$$\lambda_{max}^t\lambda_j^t\left(\frac{(2\alpha+\lambda_{max}^t+\lambda_j^t)(\lambda_{max}^t-\lambda_j^t)}{(\alpha+\lambda_{max}^t)^2(\alpha+\lambda_j^t)^2}\right) \geq (\lambda_{max}^t - \lambda_j^t)\cdot\frac{2\alpha\delta}{(\alpha+M)^4}, \tag{C.169}$$

which is justified as follows

$$\begin{aligned} \lambda_{max}^t\lambda_j^t\left(\frac{(2\alpha+\lambda_{max}^t+\lambda_j^t)(\lambda_{max}^t-\lambda_j^t)}{(\alpha+\lambda_{max}^t)^2(\alpha+\lambda_j^t)^2}\right) &= (\lambda_{max}^t-\lambda_j^t)\cdot\left(\frac{(2\alpha+\lambda_{max}^t+\lambda_j^t)\lambda_{max}^t\lambda_j^t}{(\alpha+\lambda_{max}^t)^2(\alpha+\lambda_j^t)^2}\right) \\ &\geq (\lambda_{max}^t-\lambda_j^t)\cdot\frac{2\alpha\cdot 1\cdot\delta}{(\alpha+M)^2(\alpha+M)^2}, \end{aligned}$$

where we used that $\lambda_{max}^t \geq 1$ since $\sum_{i=1}^n \lambda_i^t = n$. Hence, using the previous observation about mapping of extremes in (C.167) and the observation above, we get from (C.168) that

$$\lambda_{max}^{t+1} \leq \lambda_{max}^t - \eta\cdot\frac{1}{n}\sum_{j=1}^{n}\left[(\lambda_{max}^t-\lambda_j^t)\cdot\frac{2\alpha\delta}{(\alpha+M)^4}\right], \tag{C.170}$$

which leads to

$$\begin{aligned} \lambda_{max}^{t+1} - 1 &\leq \lambda_{max}^t - 1 - \eta\cdot\frac{1}{n}\sum_{j=1}^{n}\left[(\lambda_{max}^t-\lambda_j^t)\cdot\frac{2\alpha\delta}{(\alpha+M)^4}\right] \\ &= \lambda_{max}^t - 1 - \eta\cdot\left[(\lambda_{max}^t-1)\cdot\frac{2\alpha\delta}{(\alpha+M)^4}\right] \\ &= (\lambda_{max}^t-1)\left(1-\eta\cdot\frac{2\alpha\delta}{(\alpha+M)^4}\right) = (\lambda_{max}^t-1)(1-c\delta\cdot\eta), \end{aligned} \tag{C.171}$$

where we used that $\sum\limits_{j=1}^{n} \lambda_j^t = n$ in the first equality. Hence, using that $\lambda_{max}^t \geq 1$ as $\sum_{j=1}^n \lambda_j^t = n$ we have

$$|\lambda_{max}^{t+1} - 1| = \lambda_{max}^{t+1} - 1 \leq |\lambda_{max}^t - 1| \cdot (1 - c\delta \cdot \eta). \tag{C.172}$$

Similarly to the previous bound, we get that

$$\lambda_{min}^t \lambda_j^t \left(\frac{(2\alpha + \lambda_{min}^t + \lambda_j^t)(\lambda_{min}^t - \lambda_j^t)}{(\alpha + \lambda_{min}^t)^2 (\alpha + \lambda_j^t)^2} \right) \leq \lambda_j^t (\lambda_{min}^t - \lambda_j^t) \frac{2\alpha\delta}{(\alpha + M)^4},$$

since $\lambda_{min}^t \leq \lambda_t$. Hence, using the previous observation about mapping of extremes in (C.167) and the observation above, we deduce from (C.168) that

$$\begin{aligned} \lambda_{min}^{t+1} - 1 &\geq (\lambda_{min}^t - 1) - \eta \cdot \frac{1}{n} \sum_{j=1}^{n} \left[\lambda_j^t (\lambda_{min}^t - \lambda_j^t) \frac{2\alpha\delta}{(\alpha + M)^4} \right] \\ &= (\lambda_{min}^t - 1) - \eta \cdot \lambda_{min}^t \cdot \frac{2\alpha\delta}{(\alpha + M)^4} + \eta \cdot \frac{2\alpha\delta}{(\alpha + M)^4} \cdot \frac{1}{n} \sum_{j=1}^{t} \left(\lambda_i^t \right)^2 \\ &\geq (\lambda_{min}^t - 1) - \eta \cdot (\lambda_{min}^t - 1) \cdot \frac{2\alpha\delta}{(\alpha + M)^4} \\ &= (\lambda_{min}^t - 1) \cdot (1 - c\delta \cdot \eta), \end{aligned} \tag{C.173}$$

where in the second inequality we used Jensen's inequality for x^2 as $\sum_{j=1}^n \lambda_j^t = n$. Hence, we get the following

$$|\lambda_{min}^{t+1} - 1| = 1 - \lambda_{min}^{t+1} \leq |\lambda_{min}^t - 1| \cdot (1 - c\delta \cdot \eta), \tag{C.174}$$

since $\lambda_{min}^t \leq 1$ as $\sum_{j=1}^n \lambda_j^t = n$.

In this view, the assumptions $\lambda_{max}^t \leq M$ and $\lambda_{min}^t \geq \delta$ follow from (C.172) and (C.174) since the extremes are getting closer to one after each iteration. Recalling that by the assumption on initialization

$$\max_i |\lambda_i^0 - 1| \leq \max\{(M-1), (1-\delta)\},$$

the claim follows. □

C.6 Proofs for General Covariance

Lemma C.6.1. *Assume that $\{\hat{\gamma}_i\}_{i \in [K]}, \{\hat{s}_i\}_{i \in [K]}$ minimize*

$$-\frac{\left(\sum_{i=1}^K D_i \gamma_i\right)^2}{\left(g(1) \cdot n + \sum_{i=1}^K \frac{\gamma_i^2}{s_i}\right)}. \tag{C.175}$$

Then, for any $i < j$, we must have $\hat{s}_i = \min\{\hat{s}_i + \hat{s}_j, k_i\}$.

Proof of Lemma C.6.1. Since the $\{\hat{\gamma}_i\}_{i \in [K]}, \{\hat{s}_i\}_{i \in [K]}$ are optimal, if we fix two indices $i < j$ the corresponding $\hat{\gamma}_i, \hat{\gamma}_j, \hat{s}_i, \hat{s}_j$ are optimal among all $\gamma_i, \gamma_j, s_i, s_j$ satisfying

$$\begin{cases} 0 < \gamma_i + \gamma_j = \gamma := \hat{\gamma}_i + \hat{\gamma}_j \leq n, \\ 0 < s_i + s_j = s := \hat{s}_i + \hat{s}_j \leq \min\{n, k_i + k_j\}. \end{cases} \tag{C.176}$$

Thus, we proceed by analysing the solution for two fixed indices under the constraints (C.176) (keeping all other $\hat{\gamma}_l, \hat{s}_l$ for $l \notin \{i,j\}$ fixed). Note that, for each fixed (γ_i, γ_j) satisfying the constraints (C.176), the following objective

$$\begin{aligned} &\frac{\gamma_i^2}{s_i} + \frac{\gamma_j^2}{s_j} \to \min_{s_i, s_j} \\ &\text{s.t.} \quad s_i \le k_i, \; s_j \le k_j, \; s_i + s_j = s \end{aligned} \tag{C.177}$$

is equivalent to finding optimal ranks for (C.175). Importantly, in (C.177) we consider continuous (s_i, s_j). This relaxation has the same minimum, since we will show that the optimal s_i, s_j have integer values. We may also assume that $\gamma_j > 0$ as otherwise clearly $s_i = \min\{s, k_i\}$ is optimal.

Since (C.177) is strictly convex (on the domain given by the constraints), we can find its unique minimizer by finding a solution to the KKT conditions:

$$\begin{aligned} -\frac{\gamma_i^2}{s_i^2} + (\lambda + \mu_i) = 0, \quad -\frac{\gamma_j^2}{s_j^2} + (\lambda + \mu_j) = 0, \\ \mu_i, \mu_j \ge 0, \quad \mu_i(s_i - k_i) = 0, \quad \mu_j(s_j - k_j) = 0, \quad s = s_i + s_j. \end{aligned}$$

If $s_i = k_i$ or $s_j = 0$, then the claim is readily obtained. We will now prove that, if this is not the case, then we can find new $\widetilde{s}_i, \widetilde{s}_j, \widetilde{\gamma}_i, \widetilde{\gamma}_j$ which achieve a better value.

We first show that for $s_i < k_i, 0 < s_j < k_j$

$$\frac{\gamma_i^2}{s_i} + \frac{\gamma_j^2}{s_j} = \gamma_i \frac{\gamma}{s} + \gamma_j \frac{\gamma}{s} = \frac{\gamma^2}{s}. \tag{C.178}$$

Note that, in this case, $\mu_i = \mu_j = 0$, so the first two KKT conditions imply

$$\frac{\gamma_i}{s_i} = \sqrt{\lambda} = \frac{\gamma_j}{s_j}.$$

Thus, we have

$$\frac{\gamma_i}{s_i} = \frac{\gamma_j}{s_j} = \frac{\gamma_i + \gamma_j}{s_i + s_j} = \frac{\gamma}{s}, \tag{C.179}$$

from which (C.178) is immediate.

For the case $s_j = k_j$ and $s_i < k_i$, we have that $\mu_j \ge \mu_i = 0$, hence

$$\frac{\gamma_i}{s_i} = \sqrt{\lambda + \mu_i} \le \sqrt{\lambda + \mu_j} = \frac{\gamma_j}{s_j}.$$

From the previous case, we know that without the constraints on k_i, k_j the optimal value in (C.177) is $\frac{\gamma^2}{s}$. Thus,

$$\frac{\gamma_i^2}{s_i} + \frac{\gamma_j^2}{s_j} \ge \frac{\gamma^2}{s}.$$

Now, for $\epsilon > 0$, define $\widetilde{s}_i = s_i + \epsilon, \widetilde{s}_j = s_j - \epsilon$. Note that, as $s_i < k_i$ and $s_j > 0$, we can choose ϵ small enough such that $0 < \widetilde{s}_i < k_i, 0 < \widetilde{s}_j < k_j$. At this point, let us simply choose $\widetilde{\gamma}_i, \widetilde{\gamma}_j$ such that

$$\frac{\widetilde{\gamma}_i}{\widetilde{s}_i} = \frac{\widetilde{\gamma}_j}{\widetilde{s}_j}$$

which as in (C.178), (C.179) implies that

$$\frac{\widetilde{\gamma}_i^2}{\widetilde{s}_i}+\frac{\widetilde{\gamma}_j^2}{\widetilde{s}_j}=\frac{\gamma^2}{s}\leq\frac{\gamma_i^2}{s_i}+\frac{\gamma_j^2}{s_j}. \tag{C.180}$$

We also have $\widetilde{\gamma}_i > \gamma_i$, as otherwise

$$\frac{\widetilde{\gamma}_i}{\widetilde{s}_i}<\frac{\gamma_i}{s_i}\leq\frac{\gamma_j}{s_j}<\frac{\widetilde{\gamma}_j}{\widetilde{s}_j}$$

would be a contradiction. This gives that

$$D_i\gamma_i+D_j\gamma_j<D_i\widetilde{\gamma}_i+D_j\widetilde{\gamma}_j,$$

which implies that our new choice achieves a lower value for (C.175), thus giving the desired contradiction.

□

Lemma C.6.2. *Assume that f, f_i are differentiable strictly convex functions on $\mathbb{R}$ such that*

$$f_i'(0)<f_j'(0)<0,\ i<j,\quad \lim_{m_i\to+\infty}f_i'(m_i)=+\infty,\quad \lim_{m_i\to-\infty}f_i'(m_i)=-\infty, \tag{C.181}$$

and

$$f(0)=f'(0)=0,\quad \lim_{m\to+\infty}f'(m)=+\infty. \tag{C.182}$$

Then, the objective given by

$$\min_{m_i\geq 0} f\left(m\right)+\sum_{i=1}^{K}f_i(m_i),\quad m=\sum_i^K m_i \tag{C.183}$$

has a unique minimizer. It is uniquely characterised by being of the form $(m_1,\dots,m_M,0,\dots,0)$ and satisfying

$$m=\sum_{i=1}^{M}\left(\left(-f_i'\right)^{-1}\circ f'\right)(m),\quad m_i=\left(\left(-f_i'\right)^{-1}\circ f'\right)(m)\geq 0,\quad f'(m)+f_i'(m_i)\geq 0,\quad i\in[M]. \tag{C.184}$$

Furthermore, it can be obtained via binary search by finding the largest index M, such that the corresponding m_i are all strictly positive.

While the assumptions of this theorem might seem technical, most of them can be relaxed. However, we note that all such assumptions are fulfilled by the setting being studied and relaxing them would come at the cost of the readability of the proof of Lemma C.6.2.

Proof of Lemma C.6.2. We start by showing that (C.183) has a unique minimizer. Recall that f and f_i are strictly convex functions, and, hence, their derivatives f' and f_i' are increasing. From (C.182), we also obtain that $\lim_{m\to+\infty}f'(m)=+\infty$. By monotonicity, we have $f_i'(m_i)\geq f_i'(0)$. Therefore,

$$\lim_{m\to+\infty}f'(m)+\sum_{i=1}^{K}f_i'(m_i)=+\infty,$$

and thus

$$\lim_{m \to +\infty} f(m) + \sum_{i=1}^{K} f_i(m_i) = +\infty.$$

As a consequence, the objective achieves its infimum. Therefore, as $f(m) + \sum_{i=1}^{K} f_i(m_i)$ is strictly convex, the minimum is unique.

Notice that Slater's condition is satisfied, since the feasible set of (C.183) has an interior point. Hence, $\{m_i\}_{i=1}^{K}$ is a unique minimizer of (C.183) if and only if it satisfies the following KKT conditions (for the "if and only if" statement, see for instance page 244 in [BBV04]):

1. Stationary condition: $f'(m) + f_i'(m_i) - \lambda_i = 0$.
2. Primal feasibility: $m_i \geq 0$.
3. Complementary slackness: $\lambda_i m_i = 0$.
4. Dual feasibility: $\lambda_i \geq 0$.

In particular, the uniqueness of the minimizer implies that the KKT conditions have a unique solution. Thus, we only need to show that the m_i found by this procedure satisfy the above equations.

We now show that the active set $\mathcal{A} := \{i : m_i > 0\}$ for the optimal m_i is monotone, meaning that $\mathcal{A} = [M]$ for some $M \leq K$. We prove the statement by contradiction. Assume that there exists $m_i = 0$ and $m_j > 0$ where $i < j$. Recall that f_j' is strictly increasing, which by the ordering condition (C.181) implies that

$$f_i'(0) + f'\left(\sum_{\ell=1}^{K} m_\ell\right) < f_j'(m_j) + f'\left(\sum_{\ell=1}^{K} m_\ell\right).$$

Hence, taking some sufficiently small mass from m_j and redistributing it in m_i will decrease the objective value in (C.183), which concludes the proof.

Fix $M \leq K$. We now show that the solution of the following system of equations

$$f'(m) + f_i'(m_i) = 0, \quad \forall i \leq M \tag{C.185}$$

exists and unique. Note that this system comes from the 1. and 3. KKT conditions.

As f_i' is strictly monotone, its inverse exists and, hence, from (C.185) we get

$$m_i = (-f_i')^{-1}(f'(m)), \tag{C.186}$$

which gives

$$m = \sum_{i=1}^{M} (-f_i')^{-1}(f'(m)). \tag{C.187}$$

Let us argue the existence and uniqueness of the solution of equation (C.187) for a fixed M. Recall that f_i' is increasing and, thus, $-f_i'$ is decreasing. The inverse of a decreasing function is decreasing, hence $(-f_i')^{-1}$ is decreasing. Recalling that f' is increasing and that the composition of an increasing and a decreasing function is decreasing, it follows that $(-f_i')^{-1}(f'(m))$ is decreasing. By assumption $f_i'(0) < 0$ and f_i' is increasing such that $\lim_{m_i \to +\infty} f_i(m_i) = +\infty$, therefore the value $(-f_i')^{-1}(0)$ is well-defined and

$$(-f_i')^{-1}(0) > 0.$$

Thus, we have that

$$g_M(m) = \sum_{i=1}^{M} (-f_i')^{-1}(f'(m)) - m$$

is a strictly decreasing function with

$$\lim_{m \to +\infty} g_M(m) = -\infty, \quad g_M(0) > 0.$$

In this view, the solution of (C.187) exists and unique.

Next, we elaborate on why (C.186) is well-defined given the solution of (C.187). Note that, by our assumptions,

$$\lim_{m_i \to +\infty} f_i'(m_i) = +\infty, \quad \lim_{m_i \to -\infty} f_i'(m_i) = -\infty,$$

hence, the same holds for $(-f_i')^{-1}$, and, thus, due to continuity the quantity

$$(-f_i')^{-1}(x)$$

is well-defined for any $x \in \mathbb{R}$. Given this, we readily have that the solution of the system (C.185) exists and unique. Furthermore, this solution can be found using (C.187) and (C.186). Note also that (C.187) and (C.186) agree with (C.184).

We now show that the following procedure finds the optimal active set $\mathcal{A}^* = [M^*]$. Let $m_i(M),\ i \leq M$ be a solution of (C.185) for fixed value of $M \leq K$, and define $m(M) := \sum_{i=1}^{M} m_i(M)$. Using (C.187) and (C.186) find the smallest M such that the corresponding $m_M(M)$ is non-negative, then $M^* = M - 1$ if $M \geq 1$, otherwise, $m = m_i = 0,\ \forall i \in [K]$. If no such M was found, $M^* = [K]$. To show that the described procedure in fact gives the optimal active set $\mathcal{A}^* = [M^*]$, we need to prove that

1. If $M < M^*$, then $m_i(M) \geq 0$.
2. If $M > M^*$, then $m_M(M) \leq 0$.

Clearly, these two conditions imply that the active set of the minimizer is given by $[M^*]$, and it can be found via binary search.

We start by proving the first property. Note that, by the KKT conditions on the optimizer M^*, we have that

$$m_i(M^*) \geq 0.$$

First assume that $m(M) > m(M^*)$. By monotonicity, it follows from (C.186) that

$$m_i(M) < m_i(M^*),$$

but

$$m(M^*) = \sum_{i=1}^{M} m_i(M^*) + \sum_{i=M+1}^{M^*} m_i(M^*) \geq \sum_{i=1}^{M} m_i(M^*) > \sum_{i=1}^{M} m_i(M) = m(M),$$

where we have used that $m_i(M^*) \geq 0$, which is a contradiction. Thus, we have that $m(M) \leq m(M^*)$. Again, by (C.186) and monotonicity,

$$m_i(M) \geq m_i(M^*),$$

and, hence, all $m_i(M)$ are non-negative.

We finally argue the second property. We start by proving a weaker statement, i.e., there exists $i \geq M^* + 1$ such that $m_i(M) < 0$. Assume that $m(M) < m(M^*)$. By (C.186) and monotonicity

$$m_i(M) > m_i(M^*),$$

hence, the following holds:

$$\begin{aligned} m(M) = \sum_{i=1}^{M} m_i(M) = \sum_{i=1}^{M^*} m_i(M) + \sum_{i=M^*+1}^{M} m_i(M) > \sum_{i=1}^{M^*} m_i(M^*) + \sum_{i=M^*+1}^{M} m_i(M) \\ = m(M^*) + \sum_{i=M^*+1}^{M} m_i(M), \end{aligned}$$

which since $m(M) < m(M^*)$ implies that $\sum_{i=M^*+1}^{M} m_i(M)$ is a negative quantity. Thus, there exists $i \geq M^* + 1$ such that $m_i(M) < 0$. Assume now that $m(M) \geq m(M^*)$. Recall that only the minimizer satisfies the KKT conditions, thus

$$f'(m(M^*)) + f'_M(0) \geq 0,$$

which, as f' is increasing, implies that

$$f'(m(M)) + f'_M(0) \geq 0.$$

By construction of $m_M(M)$, we know that

$$f'(m(M)) + f'_M(m_M(M)) = 0,$$

thus, by monotonicity of f'_M we have $m_M(M) \leq 0$.

It remains to show that it suffices to check $m_M(M) \leq 0$ and not an arbitrary $m_i(M)$ for $i \geq M^* + 1$. Assume that $m_i(M) \leq 0$ for some $i \leq M$. Recall that by assumption

$$f'_i(0) < f'_M(0) < 0,$$

and by construction we have

$$f'_i(m_i(M)) = f'_M(m_M(M)) = -f'(m(M)).$$

Since f'_i is a decreasing function, we get that $-f'(m(M)) < f'_i(0)$. Recalling that $f'_i(0) < f'_M(0)$, we get $-f'(m(M)) < f'_M(0)$ and, hence, by monotonicity of f'_M we obtain that $m_M(M) \leq 0$, which concludes the proof. □

Lemma C.6.3. *The minimizer of* (5.24) *can be computed in* $\log(K)$ *steps via binary search by finding the smallest index* M^* *such that*

$$\frac{g(1)}{c_1^2 n} \sum_{j=1}^{M^*+1} s_j(D_{M^*+1} - D_j) + D_{M^*+1} \leq 0. \tag{C.188}$$

Then, the optimal active set has the form $\mathcal{A} = [M^*]$ and corresponding non-zero β_i , for $i \leq M^*$, are computed as

$$\beta_i = \frac{s_i}{c_1} \cdot \left(\frac{\frac{g(1)}{c_1^2 n} \sum_{j \in \mathcal{A}} s_j \Delta_j + D_1}{\frac{g(1)}{c_1^2 n} \sum_{j \in \mathcal{A}} s_j + 1} - \Delta_i \right), \tag{C.189}$$

where $\Delta_j = D_1 - D_j$.

Proof of Lemma C.6.3. By rescaling $g(x)$ as $\frac{g(x)}{c_1^2}$ and β_i as $c_1 \beta_i$, we may without loss of generality assume that $c_1 = 1$. From the results of Lemma C.6.2, by a direct computation, we get that for $\mathcal{A} = [M]$

$$\beta_j(M) = m_j(M) = s_j \cdot \left(\frac{\frac{g(1)}{n} \sum_{i=1}^M s_i \Delta_i + D_1}{\frac{g(1)}{n} \sum_{i=1}^M s_i + 1} - \Delta_j \right), \ \forall j \leq M,$$

thus, applying the described binary search procedure to find M^* such that $M^* + 1 = \min\left(\arg\min_M \mathbb{1}[m_M(M) > 0]\right)$ finishes the proof.

We now elaborate on the computations. For the compactness of the notation, we omit the dependence on active set in m_i's and m. We apply Lemma C.6.2 with

$$f(x) = \frac{g(1)}{n} \cdot x^2, \quad f_i(x) = \frac{x^2}{s_i} - 2D_i x,$$

which gives

$$f'(x) = \frac{2g(1)}{n} \cdot x, \quad f_i'(x) = \frac{2x}{s_i} - 2D_i.$$

Hence, we obtain that

$$(-f_i')^{-1}(x) = \frac{s_i \cdot (2D_i - x)}{2},$$

and, thus, by (C.187) we obtain

$$m = \sum_{i=1}^M (-f_i')^{-1}(f'(m)) = -f'(m) \cdot \sum_{i=1}^M \frac{s_i}{2} + \sum_{i=1}^M D_i s_i = -\frac{g(1)}{n} \cdot m \cdot \sum_{i=1}^M s_i + \sum_{i=1}^M D_i s_i.$$

In this view, we get

$$m = \frac{\sum_{i=1}^M D_i s_i}{\frac{g(1)}{n} \sum_{i=1}^M s_i + 1},$$

and, hence, since by (C.186) the following holds

$$m_j = (-f_j')^{-1}(f'(m)),$$

we get

$$\begin{aligned} m_j &= s_j \cdot \frac{2D_j - f'(m)}{2} = s_j \cdot \frac{2D_j \left(\frac{g(1)}{n} \sum_{i=1}^M s_i + 1 \right) - \frac{2g(1)}{n} \cdot \sum_{i=1}^M D_i s_i}{2 \cdot \left(\frac{g(1)}{n} \sum_{i=1}^M s_i + 1 \right)} \\ &= s_j \cdot \frac{\frac{g(1)}{n} \sum_{i=1}^M D_j s_i + D_j - \frac{g(1)}{n} \sum_{i=1}^M D_i s_i + \frac{g(1)}{n} \sum_{i=1}^M D_1 s_i - \frac{g(1)}{n} \sum_{i=1}^M D_1 s_i - D_1 + D_1}{\frac{g(1)}{n} \sum_{i=1}^M s_i + 1} \\ &= s_j \cdot \left(\frac{\frac{g(1)}{n} \sum_{i=1}^M s_i \Delta_i + D_1}{\frac{g(1)}{n} \sum_{i=1}^M s_i + 1} - \Delta_j \right), \end{aligned}$$

where $\Delta_j = D_1 - D_j$. It is easy to verify that the condition

$$\frac{g(1)}{n} \sum_{j=1}^{M^*+1} s_j(D_{M^*+1} - D_j) + D_{M^*+1} \leq 0$$

described in the statement of the lemma is equivalent to $\beta_{M^*+1}(M^*+1) = m_{M^*+1}(M^*+1) \leq 0$, which concludes the proof. □

Proof of Theorem 8. We start by showing how the lower bound reduces to the objective in (5.24). Consider the following block decomposition of $\boldsymbol{B}$ in accordance with $\boldsymbol{D}$ as in (5.29)

$$\boldsymbol{B} = [\boldsymbol{\Gamma}_1 \boldsymbol{B}_1 | \cdots | \boldsymbol{\Gamma}_K \boldsymbol{B}_K],$$

where $\boldsymbol{B}_j \in \mathbb{R}^{n \times k_j}$ with $\|(\boldsymbol{B}_j)_{i,:}\|_2 = 1$ and $\{\boldsymbol{\Gamma}_j\}_{j=1}^K$ are diagonal matrices.

Since we require $\|\boldsymbol{B}_{i,:}\|_2 = 1$, the $\boldsymbol{\Gamma}_i$ must satisfy

$$\sum_{j=1}^K \boldsymbol{\Gamma}_j^2 = \boldsymbol{I}. \tag{C.190}$$

Thus, up to a multiplicative factor $1/d$ and an additive term $\mathrm{Tr}\left[\boldsymbol{D}^2\right]$, the objective (5.23) can be written as:

$$\beta^2 \left(\mathrm{Tr}\left[\boldsymbol{M} f(\boldsymbol{M})\right]\right) - 2c_1\beta \cdot \sum_{i=1}^K D_i \cdot \mathrm{Tr}\left[\boldsymbol{\Gamma}_i^2\right], \tag{C.191}$$

where $\boldsymbol{M} = \sum_{i=1}^K \boldsymbol{M}_i := \sum_{i=1}^K \boldsymbol{\Gamma}_i \boldsymbol{B}_i \boldsymbol{B}_i^\top \boldsymbol{\Gamma}_i$. Recall that $f(x) = c_1^2 x + g(x)$, where g is the sum of odd monomials. Hence, we will be able to lower bound the terms in the first trace of (C.191) in a similar fashion to Proposition 5.4.3. Note that

$$\mathrm{Tr}\left[\boldsymbol{M}_i^2\right] = \langle \boldsymbol{1}, \boldsymbol{M}_i^{\circ 2} \boldsymbol{1} \rangle,$$

so applying Theorem A in [Kha21] gives that

$$(\boldsymbol{\Gamma}_i \boldsymbol{B}_i \boldsymbol{B}_i^\top \boldsymbol{\Gamma}_i)^{\circ 2} \succeq \frac{1}{s_i} \cdot \mathrm{Diag}(\boldsymbol{\Gamma}_i^2) \mathrm{Diag}(\boldsymbol{\Gamma}_i^2)^\top,$$

where $s_i = \mathrm{rank}(\boldsymbol{B}_i \boldsymbol{B}_i^\top)$. Thus, we have the bound

$$\mathrm{Tr}\left[\boldsymbol{M}_i^2\right] \geq \frac{1}{s_i} \left(\mathrm{Tr}\left[\boldsymbol{\Gamma}_i^2\right]\right)^2$$

Since $xg(x) \geq 0$, we can lower bound the rest of the terms with the identity, i.e.,

$$\mathrm{Tr}\left[\boldsymbol{M} g(\boldsymbol{M})\right] = \langle \boldsymbol{1}, \boldsymbol{M} \circ g(\boldsymbol{M}) \boldsymbol{1} \rangle \geq g(1) \cdot n$$

as $\mathrm{Diag}(\boldsymbol{M}) = \boldsymbol{I}$. Consequently, neglecting the cross-terms $\mathrm{Tr}\left[\boldsymbol{M}_i \boldsymbol{M}_j\right]$ (as the trace of the product of PSD matrices is non-negative) we arrive at

$$\mathrm{Tr}\left[\boldsymbol{M} f(\boldsymbol{M})\right] \geq g(1) \cdot n + c_1^2 \cdot \sum_{i=1}^K \frac{1}{s_i} \left(\mathrm{Tr}\left[\boldsymbol{\Gamma}_i^2\right]\right)^2.$$

Defining $\gamma_i := \mathrm{Tr}\left[\boldsymbol{\Gamma}_i^2\right] \geq 0$, we arrive at the following lower bound on (C.191):

$$\beta^2 \left(g(1) \cdot n + \sum_{i=1}^K \frac{\gamma_i^2}{s_i}\right) - 2\beta \cdot \sum_{i=1}^K D_i \gamma_i, \tag{C.192}$$

where, with an abuse of notation, we rescale $g(1) := g(1)/c_1^2$ and $\beta := c_1\beta$. Now, by choosing $\beta_i := \beta\gamma_i$ and using that $\sum_{i=1}^K \gamma_i = n$ due to (C.190), the objective (C.192) is seen to be equivalent to (5.24). This shows that (5.23) $\geq \mathrm{LB}(\boldsymbol{D})$. We now give a brief outline of how one can obtain the optimal s_i and β_i for (5.24).

For finding the optimal s_i, it is more natural to still consider (C.192). Due to the block form (5.29), the s_i have to satisfy the constraints in (5.25). Note that (C.192) evaluated at the optimal β is equal to

$$\text{(C.192)} \geq -\frac{\left(\sum_{i=1}^K D_i\gamma_i\right)^2}{\left(g(1)\cdot n + \sum_{i=1}^K \frac{\gamma_i^2}{s_i}\right)}. \tag{C.193}$$

The optimal s_i for this objective are *water-filled*, i.e.,

$$\begin{cases} \boldsymbol{s} = [n, 0, \cdots, 0], & n \leq k_1, \\ \boldsymbol{s} = [k_1, k_2, \cdots, k_K], & d \leq n, \\ \boldsymbol{s} = [k_1, \cdots, k_{\mathrm{id}(n)-1}, \mathrm{res}(n), 0, \cdots, 0] & \text{otherwise}, \end{cases} \tag{C.194}$$

where $\boldsymbol{s} = [s_1, \cdots, s_k]$ and $\mathrm{id}(n)$ denotes the first position at which

$$\min\{n, d\} - \sum_{i=1}^{\mathrm{id}(n)} k_i < 0,$$

and

$$\mathrm{res}(n) = \min\{n, d\} - \sum_{i=1}^{\mathrm{id}(n)-1} k_i.$$

This follows directly from Lemma C.6.1. It only remains to show that the optimal β_i can be obtained via (5.28), which is done in Lemma C.6.3. This concludes the proof. □

Proof of Proposition 5.5.2. Except for terms of the form $\mathrm{Tr}\left[\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{B}_j\boldsymbol{B}_j^\top\right]$, all the other terms can be estimated as in the proof of Proposition 5.4.3. The only technical difference is that all the constants now depend on the ratios $\frac{k_i}{n}$.

We will show that, with probability at least $1 - c\exp\left(-cd^\epsilon\right)$, for all $i \neq j$,

$$\mathrm{Tr}\left[\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{B}_j\boldsymbol{B}_j^\top\right] \leq n^{\frac{1}{2}+\epsilon}. \tag{C.195}$$

Thus, by a simple union bound, we have that, with probability at least $1 - \frac{c}{d^2}$, this bound holds jointly for all pairs $\boldsymbol{B}_i, \boldsymbol{B}_j$. It follows as in the proof of Lemma C.2.3 that we can write

$$\boldsymbol{B}_i\boldsymbol{B}_i^\top = \boldsymbol{P}_i\boldsymbol{U}\boldsymbol{D}_i\boldsymbol{U}^\top\boldsymbol{P}_i,$$

where by abuse of notation we pushed the factor $\frac{n}{k_i}$ in $\boldsymbol{D}_i$ (which will only affect the constants c, C). Here, $\boldsymbol{P}_i$ is a diagonal matrix such that, for any $\epsilon > 0$, with probability at least $1 - c\exp\left(-cd^\epsilon\right)$, we have that

$$\left\|\boldsymbol{P}_i - \boldsymbol{I}\right\|_{op} \leq n^{-\frac{1}{2}+\epsilon}.$$

To see this, first observe that $\Theta : (\mathbb{R}^{n\times n})^4 \mapsto \mathbb{R}$ given by

$$\Theta(\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3, \boldsymbol{X}_4) = \mathrm{Tr}\left[\boldsymbol{X}_1\boldsymbol{U}\boldsymbol{D}_i\boldsymbol{U}^\top\boldsymbol{X}_2\boldsymbol{X}_3\boldsymbol{U}\boldsymbol{D}_j\boldsymbol{U}^\top\boldsymbol{X}_4\right]$$

is differentiable (as it is the composition of the trace function with 4-linear form). Since by construction

$$\mathrm{Tr}\left[\boldsymbol{U}\boldsymbol{D}_i\boldsymbol{U}^\top\boldsymbol{U}\boldsymbol{D}_j\boldsymbol{U}^\top\right] = \mathrm{Tr}\left[\boldsymbol{0}\right] = 0,$$

this implies that, with probability at least $1 - \frac{c}{d^2}$,

$$\begin{aligned}
0 &\le \mathrm{Tr}\left[\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{B}_j\boldsymbol{B}_j^\top\right] \\
&= \mathrm{Tr}\left[\boldsymbol{P}_i\boldsymbol{U}\boldsymbol{D}_i\boldsymbol{U}^\top\boldsymbol{P}_i\boldsymbol{P}_j\boldsymbol{U}\boldsymbol{D}_j\boldsymbol{U}^\top\boldsymbol{P}_j\right] \\
&= \mathrm{Tr}\left[\boldsymbol{P}_i\boldsymbol{U}\boldsymbol{D}_i\boldsymbol{U}^\top\boldsymbol{P}_i\boldsymbol{P}_j\boldsymbol{U}\boldsymbol{D}_j\boldsymbol{U}^\top\boldsymbol{P}_j\right] - \mathrm{Tr}\left[\boldsymbol{U}\boldsymbol{D}_i\boldsymbol{U}^\top\boldsymbol{U}\boldsymbol{D}_j\boldsymbol{U}^\top\right] \\
&\le Cnn^{-\frac{1}{2}+\epsilon},
\end{aligned}$$

where in the last step we used that the derivative of the trace function is bounded by $n \cdot \left\|\cdot\right\|_{op}$. Thus, (C.195) holds.

By construction, the sum of all the cross terms is of the form

$$\sum_{i\neq j}\mathrm{Tr}\left[\boldsymbol{M}_i\boldsymbol{M}_j\right],$$

where $\boldsymbol{M}_i = \boldsymbol{\Gamma}_i\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{\Gamma}_i$, $\boldsymbol{\Gamma}_i^2 = \frac{\gamma_i}{n}\boldsymbol{I}$ and $\sum_{i=1}^K\gamma_i = n$. We have

$$\begin{aligned}
\left|\sum_{i\neq j}\mathrm{Tr}\left[\boldsymbol{M}_i\boldsymbol{M}_j\right]\right| &= \left|\sum_{i\neq j}\frac{\gamma_i\gamma_j}{n^2}\mathrm{Tr}\left[\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{B}_j\boldsymbol{B}_j^\top\right]\right| \\
&\le \sum_{i\neq j}\frac{\gamma_i\gamma_j}{n^2}\left|\mathrm{Tr}\left[\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{B}_j\boldsymbol{B}_j^\top\right]\right| \\
&\le C\sum_{i\neq j}\frac{\gamma_i\gamma_j}{n^2}n^{\frac{1}{2}+\epsilon} \\
&\le Cn^{\frac{1}{2}+\epsilon},
\end{aligned}$$

where in the third step we used a union bound on (C.195) and in the last step we used $\sum_{i=1}^K\frac{\gamma_i}{n} = 1$. □

C.7 Details of Experiments and Additional Numerical Results

We first describe the training details and the whitening procedure that is used to preprocess natural images for MNIST (Figure C.3) and CIFAR-10 (Figures 5.1, C.1 and C.2). Next, we give some remarks about the experiments concerning VAMP (Figure 5.3) and about the discontinuous behaviour of the derivative of the lower bound highlighted in Figure 5.2. In addition, we present additional numerical experiments which cover extra classes of natural images.

Activation function and weight parameterization. Note that the derivative of the sign activation is zero almost everywhere (except one point, which is the origin). In this view, we cannot use conventional gradient-based algorithms to find the optimal set of parameters for an autoencoder with the sign activation. We tackle this issue by using a straight-through estimator (see, for instance, [YLZ+19]) of the sign activation. During the forward pass the activations of the first layer are computed for $\sigma(x) = \mathrm{sign}(x)$, while during the backward pass

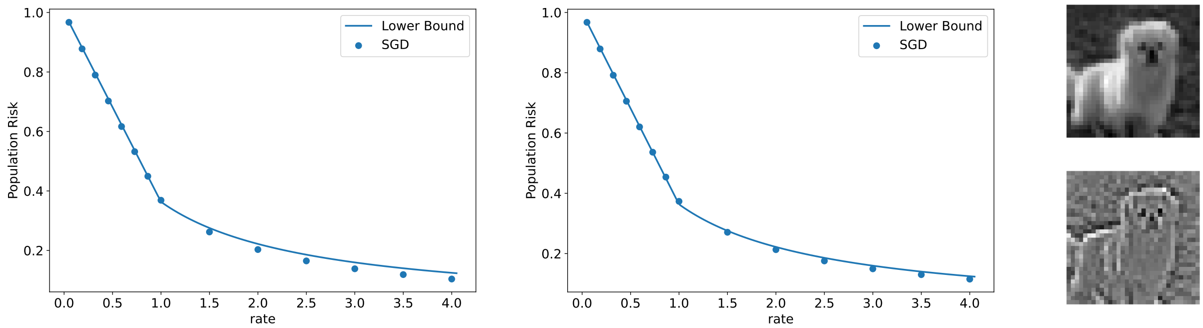


Figure C.1: Compression ($\sigma \equiv \mathrm{sign}$) of the CIFAR-10 "dog" class with a two-layer autoencoder. The data is *whitened* so that $\boldsymbol{\Sigma} = \boldsymbol{I}$: on top, an example of a grayscale image; on the bottom, the corresponding whitening. The blue dots are the population risk obtained via SGD, and they agree well with the solid line corresponding to the lower bounds of Theorem 5 and Proposition 5.4.2. Here, the effect of the number of augmentations used per image is shown. For the left plot each image was augmented 10 times, while for the right plot each image was augmented 15 times.

$\sigma(x) = \tanh(x/\tau)$ is used. Here, the temperature parameter $\tau > 0$ controls how well the differentiable surrogate $\tanh(x/\tau)$ approximates $\mathrm{sign}(x)$, as

$$\lim_{\tau \to 0} \tanh(x/\tau) = \mathrm{sign}(x), \quad \forall x \in \mathbb{R} \setminus \{0\}.$$

More precisely, the differentiable approximation becomes more accurate for smaller values of τ. However, we also note that extremely small values of τ might cause numerical issues, since the derivative of the differentiable surrogate diverges at the origin as $\tau \to 0$. For the numerical experiments, we pick $\tau \in [0.01, 0.2]$, with the exact value depending on the specific setting.

Note that the constraint on the encoder weights $\|\boldsymbol{B}_{i,:}\|_2 = 1$ can be enforced via a simple reparameterization that forces the rows of $\boldsymbol{B}$ to lie on the unit sphere $\mathbb{S}^{d-1}$. More precisely, we use the following classical differentiable reparameterization of $\boldsymbol{B}^\top = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n]$, where $\boldsymbol{b}_i = \frac{\hat{\boldsymbol{b}}_i}{\|\hat{\boldsymbol{b}}_i\|_2}$, with $\{\hat{\boldsymbol{b}}_i\}_{i=1}^n$ being the trainable parameters. We note that it is not clear a priori whether we need to impose the constraints directly for the straight-through estimator, since during the forward pass we use the norm-agnostic sign function.

Augmentation and whitening. For the experiments on natural images, we augment the data of each class 15 times. This is done to emulate the optimization of the population risk, since the amount of initial data (approximately 5000 samples per class) leads to a gap between empirical and population risks, especially for high rates. The effect of the data augmentation is represented in Figure C.1 for a whitened CIFAR-10 class. It can be seen that a mild amount of augmentation, i.e., $\times 10$ and $\times 15$, is already enough for our purposes, and the difference between the two plots is rather small. Notably, this amount of augmentation brings the dataset to the scale of the original data when all classes are considered (around 50000 training examples).

The whitening procedure used in the experiments concerning isotropic data is performed as follows: given the *centered* augmented data $\boldsymbol{X} \in \mathbb{R}^{\mathrm{n_{samples}} \times d}$, we compute its empirical covariance matrix given by

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{\mathrm{n_{samples}} - 1} \cdot \sum_{i=1}^{\mathrm{n_{samples}}} \boldsymbol{X}_{i,:} \boldsymbol{X}_{i,:}^\top,$$

and then we multiply each input by the inverse square root of it, i.e.,

$$\hat{\boldsymbol{X}}_{i,:} = \hat{\boldsymbol{\Sigma}}^{-\frac{1}{2}} \boldsymbol{X}_{i,:}.$$

The resulting whitened images are represented in Figures 5.1, C.1 and C.3.

In the experiments concerning non-isotropic data (Figures 5.1 (right plot) and C.4), we center the data with the empirical mean and divide by a *scalar* empirical variance computed across all the pixels, which is the standard preprocessing procedure widely used for computer vision tasks.

VAMP experiments. For the VAMP experiments, we implement the State Evolution (SE) recursion which exactly characterizes the limiting performance of VAMP as $d \to \infty$, see [SRF16, RSF19] for an overview. We then plot the fixed point of said SE recursion. A concrete description for VAMP is provided by Algorithm 2 in [FRS18], which however covers a more general multi-layer setting.

"Jumps" of the lower bound derivative. The derivative switch described in Figure 5.2 does not necessarily happen precisely at the point when the block is filled. A switch may occur at a later point since, even if $s_i > 0$, the corresponding optimal β_i may be 0. Intuitively, this phenomenon occurs in cases when it is still better to put more mass in the block where the rank is utilized to the fullest $(s_j = k_j)$. This corresponds to the following condition on the derivatives of the objective (5.24):

$$\frac{\partial \text{(5.24)}}{\partial \beta_i}(0) > \frac{\partial \text{(5.24)}}{\partial \beta_j}(\beta_j^*),$$

where β_i^* stands for the optimal β_i and j denotes the first index at which $\beta_j^* > 0$. This behaviour occurs when the spectrum $\boldsymbol{D}$ has a large variation in scale, e.g.,

$$\boldsymbol{D} = [5, 0.02, 0.01].$$

In this case, the last components will be utilized for n significantly larger than k_1 $(n = k_1$ precisely characterizes the point where the rank of the first block of $\boldsymbol{B}$, i.e., $\boldsymbol{B}_1$, is the maximum possible). Note that, for this choice of $\boldsymbol{D}$, the plot of the derivative analogous to Figure 5.2 will not indicate such prominent "jumps". In fact, the contribution of the last components to the derivative value is less significant in comparison to the analogous quantity evaluated for the top-most eigenvalues.

Additional experimental data. We also provide additional numerical simulations, similar to those presented in the body of the paper. In particular, we provide more class variations for the natural data experiments (MNIST and CIFAR-10).

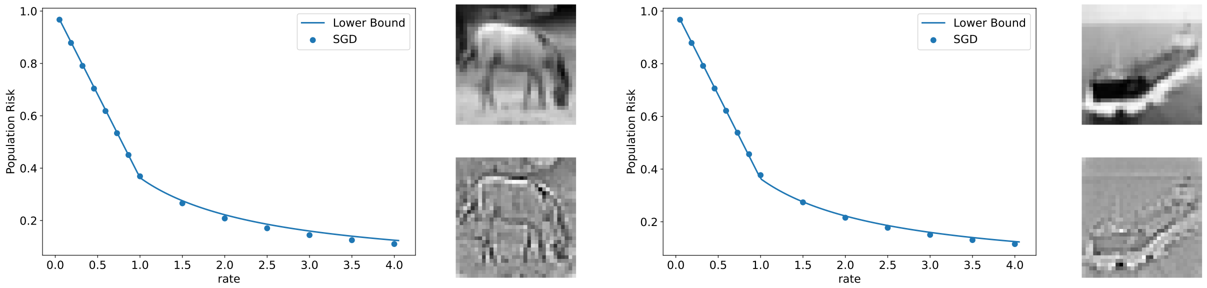


Figure C.2: Compression ($\sigma \equiv \mathrm{sign}$) of the CIFAR-10 "horse" class (left) and "ship" class (right) with a two-layer autoencoder. The data is *whitened* so that $\boldsymbol{\Sigma} = \boldsymbol{I}$: on top, an example of a grayscale image; on the bottom, the corresponding whitening. The blue dots are the population risk obtained via SGD, and they agree well with the solid line corresponding to the lower bounds of Theorem 5 and Proposition 5.4.2. Here, in both cases the amount of augmentations per image is equal to 15.

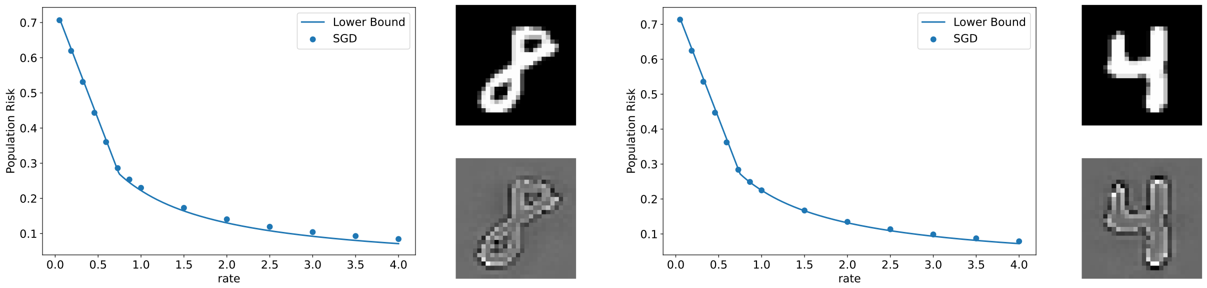


Figure C.3: Compression ($\sigma \equiv \mathrm{sign}$) of the MNIST "8" class (left) and "4" class (right) with a two-layer autoencoder. The data is *whitened* so that $\boldsymbol{\Sigma} = \boldsymbol{I}$: on top, an example of a grayscale image; on the bottom, the corresponding whitening. The blue dots are the population risk obtained via SGD, and they agree well with the solid line corresponding to the lower bounds of Theorem 5 and Proposition 5.4.2. Here, in both cases the amount of augmentations per image is equal to 10.

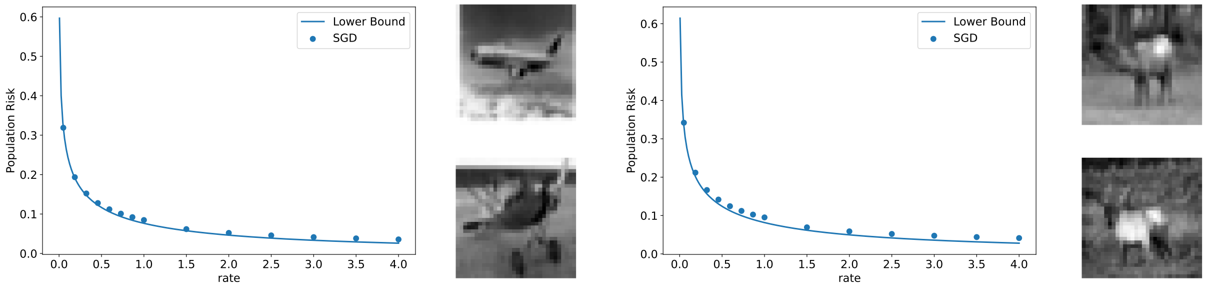


Figure C.4: Compression ($\sigma \equiv \mathrm{sign}$) of the CIFAR-10 "airplane" class (left) and "deer" class (right) with a two-layer autoencoder. The data is *not whitened* ($\boldsymbol{\Sigma} \neq \boldsymbol{I}$). The blue dots are the SGD population risk, and they are close to the lower bound of Theorem 8. Here, in both cases the amount of augmentations per image is equal to 15.

APPENDIX D

Appendix for Chapter 6

D.1 MSE Characterizations

D.1.1 Proof of Proposition 6.4.1

Denote by $\boldsymbol{x}^1$ the first iterate of the RI-GAMP algorithm [VKM22], as in (6.20). Then, by taking σ to be the sign, one can readily verify that

$$\boldsymbol{x}^1 = \boldsymbol{B}^\top \mathrm{sign}(\boldsymbol{B}\boldsymbol{x}).$$

Note that $\boldsymbol{B}$ is bi-rotationally invariant in law and, as $\boldsymbol{x}$ has i.i.d. components, its empirical distribution converges in Wasserstein-2 distance to a random variable whose law is that of the first component of $\boldsymbol{x}$, denoted by x_1. Therefore, the assumptions of Theorem 3.1 in [VKM22] are satisfied. Hence, for any ψ pseudo-Lipschitz of order 2,[1] we have that, almost surely,

$$\lim_{d\to\infty} \frac{1}{d} \sum_{i=1}^{d} \psi((\boldsymbol{x}^1)_i, (\boldsymbol{x})_i) = \mathbb{E}[\psi(\mu x_1 + \sigma g, x_1)],$$

where $g \sim \mathcal{N}(0,1)$ is independent of x_1 and the state evolution parameters (μ, σ) for $r \leq 1$ can be computed as

$$\mu = r \cdot \sqrt{\frac{2\kappa_2}{\pi}} = r \cdot \sqrt{\frac{2}{\pi}}, \quad \sigma^2 = r \cdot \left(\kappa_2 + \kappa_4 \cdot \frac{2}{\pi\kappa_2}\right) = r \cdot \left(1 - r \cdot \frac{2}{\pi}\right), \tag{D.1}$$

that is equation (11) in [VKM22]. Here, $\{\kappa_{2k}\}_{k\in\mathbb{N}}$ denote the rectangular free cumulants of the constant random variable equal to 1 (since all the singular values of $\boldsymbol{B}$ are equal to 1 by assumption). Noting that $\psi(x,y) = (x - \alpha \cdot y)^2$ is pseudo-Lipschitz of order 2, we get that, almost surely,

$$\lim_{d\to\infty} \frac{1}{d} \cdot \|\boldsymbol{x} - \alpha \cdot \boldsymbol{B}^\top \mathrm{sign}(\boldsymbol{B}^\top \boldsymbol{x})\|_2^2 = \mathbb{E}_{x_1,g}[|x_1 - \alpha(\mu x_1 + \sigma g)|_2^2],$$

which implies that

$$\lim_{d\to\infty} \frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}}\|\boldsymbol{x} - \alpha \cdot \boldsymbol{B}^\top \mathrm{sign}(\boldsymbol{B}^\top \boldsymbol{x})\|_2^2 = \mathbb{E}_{x_1,g}[|x_1 - \alpha(\mu x_1 + \sigma g)|_2^2].$$

[1] We recall that $\psi : \mathbb{R}^2 \to \mathbb{R}$ is pseudo-Lipschitz of order 2 if, for all $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^2$, $|\psi(\boldsymbol{a}) - \psi(\boldsymbol{b})| \leq L\|\boldsymbol{a} - \boldsymbol{b}\|_2(1 + \|\boldsymbol{a}\|_2 + \|\boldsymbol{b}\|_2)$ for some constant $L > 0$.

By expanding the RHS of the last equation and using that x_1 has unit second moment by assumption, we get

$$\begin{aligned}\mathbb{E}_{x_1,g}[|x_1 - \alpha(\mu x_1 + \sigma g)|_2^2] &= (1-\alpha\mu)^2 \cdot \mathbb{E}[x_1^2] + \alpha^2\sigma^2 \cdot \mathbb{E}[g^2] = (1-\alpha\mu)^2 + \alpha^2\sigma^2 \\ &= 1 - 2\alpha\mu + \alpha^2(\mu^2+\sigma^2) = 1 - 2\alpha \cdot r\sqrt{\frac{2}{\pi}} + \alpha^2 r.\end{aligned}$$

Thus, by minimizing over α, we have

$$\min_\alpha \mathbb{E}_{x_1,g}[|x - \alpha(\mu x + \sigma g)|_2^2] = 1 - \frac{2}{\pi} \cdot r,$$

which concludes the proof of (6.11).

To prove (6.12), a direct calculation gives

$$\begin{aligned}\frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \left\| \boldsymbol{x} - \alpha \cdot \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0}_{(d-n)\times n} \end{bmatrix} \operatorname{sign}([\boldsymbol{I}_n, \boldsymbol{0}_{n\times(d-n)}]\boldsymbol{x}) \right\|_2^2 &= 1 - r + r \cdot \mathbb{E}[(x_1 - \alpha \operatorname{sign}(x_1))^2] \\ &= 1 - r + r \cdot (\mathbb{E}[x_1^2] - 2\alpha \cdot \mathbb{E}[|x_1|] \\ &\quad + \alpha^2 \cdot \mathbb{E}[\operatorname{sign}^2(x_1)]) \\ &= 1 - r + r \cdot (1 - 2\alpha \cdot \mathbb{E}[|x_1|] + \alpha^2) \\ &= 1 + r \cdot (\alpha^2 - 2\alpha \cdot \mathbb{E}[|x_1|]).\end{aligned}$$

The RHS is minimized by $\alpha = \mathbb{E}[|x_1|]$, which gives

$$\min_\alpha \frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \left\| \boldsymbol{x} - \alpha \cdot \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0}_{(d-n)\times n} \end{bmatrix} \operatorname{sign}([\boldsymbol{I}_n, \boldsymbol{0}_{n\times(d-n)}]\boldsymbol{x}) \right\|_2^2 = 1 - r \cdot (\mathbb{E}[|x_1|])^2,$$

and the proof is complete. □

D.1.2 Proof of Proposition 6.5.1

Let $\hat{\boldsymbol{x}}^1$ be an iterate of the RI-GAMP algorithm [VKM22], as in (6.20). Then, by taking σ to be the sign and $f_t = f$, one can readily verify that

$$\hat{\boldsymbol{x}}^1 = f(\boldsymbol{B}^\top \operatorname{sign}(\boldsymbol{B}\boldsymbol{x})),$$

which is exactly the form of the autoencoder in (6.4) that we wish to analyze. Thus, as f is Lipschitz, the assumptions of Theorem 3.1 in [VKM22] are satisfied and, following the same passages as in the proof of Proposition 6.4.1, we have

$$\lim_{d\to\infty} \frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \|\boldsymbol{x} - f(\boldsymbol{B}^\top \operatorname{sign}(\boldsymbol{B}\boldsymbol{x}))\|_2^2 = \mathbb{E}_{x_1,g}[|x_1 - f(\mu x_1 + \sigma g)|_2^2], \tag{D.2}$$

where x_1 is the first entry of $\boldsymbol{x}$, $g \sim \mathcal{N}(0,1)$ is independent of x_1, and (μ, σ) are given by (D.1) (which coincides with (6.17)). This concludes the proof. □

D.1.3 Proof of Proposition 6.5.2

A direct calculation gives

$$\begin{aligned}\frac{1}{d} \cdot \mathbb{E}_{\boldsymbol{x}} \left\| \boldsymbol{x} - f\left(\begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0}_{(d-n)\times n} \end{bmatrix} \operatorname{sign}([\boldsymbol{I}_n, \boldsymbol{0}_{n\times(d-n)}]\boldsymbol{x}) \right) \right\|_2^2 &= (1-r) \cdot \mathbb{E}\left[(x_1 - f(0))^2\right] \\ &\quad + r \cdot \mathbb{E}\left[(x_1 - f(\operatorname{sign}(x_1)))^2\right],\end{aligned} \tag{D.3}$$

where x_1 is the first entry of $\boldsymbol{x}$. The first term in (D.3) is minimized when $f(0) = \mathbb{E}[x] = 0$. Hence, we obtain that, at the optimum,

$$(1-r) \cdot \mathbb{E}\left[(x_1 - f(0))^2\right] = 1 - r,$$

as $\mathbb{E}[x^2] = 1$. As for the second term in (D.3), we rewrite

$$\mathbb{E}\left[(x_1 - f(\mathrm{sign}(x_1)))^2\right] = \mu_{x_1}(\{0\}) \cdot \frac{1}{2} \cdot (f(1)^2 + f(-1)^2) + \mathbb{E}[\mathbb{1}_{x_1>0}(x_1 - f(1))^2] + \mathbb{E}[\mathbb{1}_{x_1<0}(x_1 - f(-1))^2], \tag{D.4}$$

where μ_{x_1} stands for the measure that corresponds to the distribution of x_1, and we use that $\mathrm{sign}(0)$ is a Rademacher random variable by convention. As the distribution of x_1 is the same as that of $-x_1$, (D.4) is minimized by taking $f(1) = -f(-1)$. Thus, we have that

$$\min_f \text{(D.4)} = \min_{u \in \mathbb{R}} \mathbb{E}[(x_1 - u \cdot \mathrm{sign}(x_1))^2].$$

The RHS of this last expression can be further rewritten as

$$\min_{u \in \mathbb{R}} \mathbb{E}[(x_1 - u \cdot \mathrm{sign}(x_1))^2] = \mathbb{E}[x_1^2] + \min_{u \in \mathbb{R}} \left\{u^2 - 2u \cdot \mathbb{E}|x_1|\right\} = 1 - (\mathbb{E}|x_1|)^2,$$

which concludes the proof. □

D.1.4 Computation of f^*

Sparse Gaussian. Using Bayes rule, the conditional expectation can be expressed as

$$\mathbb{E}[x|\mu x + \sigma g = y] = \frac{\mathbb{E}_x\left[x \cdot P(\mu x + \sigma g = y|x)\right]}{\mathbb{E}_x\left[P(\mu x + \sigma g = y|x)\right]} = \frac{\mathbb{E}_x\left[x \cdot P(\mu x + \sigma g = y|x)\right]}{P(\mu x + \sigma g = y)}. \tag{D.5}$$

Given that $x \sim \mathrm{SG}_1(p)$, with probability p we have that $\mu x + \sigma g \sim \mathcal{N}(0, \mu^2/p + \sigma^2)$ as $x \sim \mathcal{N}(0, 1/p)$, and with probability $(1-p)$ we have that $x = 0$, and, hence, $\mu x + \sigma g = \sigma g \sim \mathcal{N}(0, \sigma^2)$. Combining gives

$$P(\mu x + \sigma g = y) = p \cdot \frac{\sqrt{p}}{\sqrt{2\pi(\mu^2 + p\sigma^2)}} \cdot \exp\left(-\frac{py^2}{2(\mu^2 + p\sigma^2)}\right) + (1-p) \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right).$$

Note that due to sparsity, we have that

$$\mathbb{E}_x\left[x \cdot P(\mu x + \sigma g = y|x)\right] = p \cdot \mathbb{E}_{x \sim \mathcal{N}(0,1/p)}\left[x \cdot P(\mu x + \sigma g = y|x)\right], \tag{D.6}$$

and, in this case, we conclude that

$$\mu x + \sigma g | x \sim \mathcal{N}(\mu x, \sigma^2).$$

Thus, the RHS of (D.6) is a Gaussian integral, which is straight-forward to calculate by "completing a square". The computation gives

$$\mathbb{E}_{x \sim \mathcal{N}(0,1/p)}\left[x \cdot P(\mu x + \sigma g = y|x)\right] = \sqrt{\frac{p}{2\pi}} \cdot \mu y \cdot \exp\left(-\frac{py^2}{2(\mu^2 + p\sigma^2)}\right) \cdot \frac{1}{(\mu^2 + p\sigma^2)^{3/2}}.$$

Note that, when $p = 1$, i.e., $\boldsymbol{x}$ is an isotropic Gaussian vector, f^* is just a rescaling by a constant factor, i.e., $f^*(y) = \mathrm{const}(\mu, \sigma) \cdot y$.

Sparse Laplace. The sparse Laplace distribution with sparsity level $(1-p)$ has the following law

$$(1-p)\cdot\delta_0 + p\cdot\sqrt{\frac{p}{2}}\cdot\exp\left(-\sqrt{2p}\cdot|x|\right), \tag{D.7}$$

where δ_0 stands for the delta distribution centered at 0. The scaling for different p is chosen to ensure a unit second moment.

First, we derive the expression for the conditional expectation for $p=1$. For $p\neq 1$ we elaborate later how a simple change of variables allows to obtain closed-form expressions of the corresponding expectations via the case $p=1$. For $p=1$, the denominator in (D.5) is equivalent to

$$\int_{\mathbb{R}} p(x)p(\mu x+\sigma g=y|x)\mathrm{d}x = \frac{1}{\sqrt{4\pi\sigma^2}}\int_{\mathbb{R}}\exp\left(-\sqrt{2}\cdot|x|\right)\exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right)\mathrm{d}x. \tag{D.8}$$

By considering two cases, i.e., $x<0$ and $x\geq 0$, for the limits of integration and for each of them "completing a square", we obtain

$$\begin{aligned}
&\int_{\mathbb{R}_+}\exp\left(-\sqrt{2}\cdot x\right)\exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right)\mathrm{d}x\\
&\qquad=\left(1+\operatorname{erf}\left(\frac{\sqrt{2}\mu y-2\sigma^2}{2\mu\sigma}\right)\right)\cdot\exp\left(\frac{\sigma^2-\sqrt{2}\mu y}{\mu^2}\right)\cdot\sqrt{\frac{\pi}{2}}\cdot\frac{\sigma}{\mu},\\
&\int_{\mathbb{R}_-}\exp\left(\sqrt{2}\cdot x\right)\exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right)\mathrm{d}x\\
&\qquad=\operatorname{erfc}\left(\frac{\sqrt{2}\mu y+2\sigma^2}{2\mu\sigma}\right)\cdot\exp\left(\frac{\sigma^2+\sqrt{2}\mu y}{\mu^2}\right)\cdot\sqrt{\frac{\pi}{2}}\cdot\frac{\sigma}{\mu},
\end{aligned}$$

where $\operatorname{erf}(\cdot)$ stands for the Gaussian error function, and $\operatorname{erfc}(\cdot)$ for its complement. For the case of $p\neq 1$, we get that the RHS of (D.8) becomes

$$(1-p)\cdot\frac{1}{\sqrt{2\pi\sigma^2}}\cdot\exp\left(-\frac{y^2}{2\sigma^2}\right)+p\cdot\sqrt{\frac{p}{4\pi\sigma^2}}\cdot\int_{\mathbb{R}}\exp\left(-\sqrt{2p}\cdot|x|\right)\exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right)\mathrm{d}x.$$

The change in normalization constant of the second term is then trivial. For the integral itself, consider the change of variables $\tilde{x}=x\cdot\sqrt{p}$:

$$\begin{aligned}
&\int_{\mathbb{R}}\exp\left(-\sqrt{2p}\cdot|x|\right)\exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right)\mathrm{d}x\\
&\qquad=\frac{1}{\sqrt{p}}\cdot\int_{\mathbb{R}}\exp\left(-\sqrt{2}\cdot|\tilde{x}|\right)\exp\left(-\frac{(y-\frac{\mu}{\sqrt{p}}\cdot\tilde{x})^2}{2\sigma^2}\right)\mathrm{d}\tilde{x}\\
&\qquad=\frac{1}{\sqrt{p}}\cdot\int_{\mathbb{R}}\exp\left(-\sqrt{2}\cdot|\tilde{x}|\right)\exp\left(-\frac{(y-\tilde{\mu}\cdot\tilde{x})^2}{2\sigma^2}\right)\mathrm{d}\tilde{x},
\end{aligned}$$

which is exactly the previous integral in (D.8) but with $\tilde{\mu}=\mu/\sqrt{p}$ and an additional scaling factor in front.

Consider the numerator of (D.5) for $p=1$. For this case, the computation reduces to evaluating:

$$\int_{\mathbb{R}} x\cdot p(x)p(\mu x+\sigma g=y|x)\mathrm{d}x = \frac{1}{\sqrt{4\pi\sigma^2}}\int_{\mathbb{R}} x\cdot\exp\left(-\sqrt{2}\cdot|x|\right)\exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right)\mathrm{d}x. \tag{D.9}$$

Reducing to cases again and “completing a square” gives

$$\begin{aligned}
&\int_{\mathbb{R}_+} x \cdot \exp\left(-\sqrt{2} \cdot x\right) \exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right) \mathrm{d}x \\
&= \exp\left(-\frac{y^2}{2\sigma^2}\right) \cdot \left[\frac{\sigma^2}{\mu^2} + \frac{\sqrt{\pi}\sigma \cdot (\sqrt{2}\mu y - 2\sigma^2) \cdot e^{\frac{(\mu y - \sqrt{2}\sigma^2)^2}{2\mu^2\sigma^2}} \cdot \left(1 + \operatorname{erf}\left(\frac{y}{\sqrt{2}\sigma} - \frac{\sigma}{\mu}\right)\right)}{2\mu^3}\right], \\
&\int_{\mathbb{R}_-} x \cdot \exp\left(\sqrt{2} \cdot x\right) \exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right) \mathrm{d}x \\
&= \exp\left(-\frac{y^2}{2\sigma^2}\right) \cdot \left[-\frac{\sigma^2}{\mu^2} + \frac{\sqrt{\pi}\sigma \cdot (\sqrt{2}\mu y + 2\sigma^2) \cdot e^{\frac{(\mu y + \sqrt{2}\sigma^2)^2}{2\mu^2\sigma^2}} \cdot \operatorname{erfc}\left(\frac{y}{\sqrt{2}\sigma} + \frac{\sigma}{\mu}\right)}{2\mu^3}\right].
\end{aligned}$$

The derivation for the case $p \neq 1$ can be obtained analogously, by noting that (D.9) in this case is written as

$$p \cdot \sqrt{\frac{p}{4\pi\sigma^2}} \cdot \int_{\mathbb{R}} x \cdot \exp\left(-\sqrt{2p} \cdot |x|\right) \exp\left(-\frac{(y-\mu x)^2}{2\sigma^2}\right) \mathrm{d}x.$$

Sparse Rademacher. The sparse Rademacher distribution with sparsity level $(1-p)$ has the following law

$$(1-p) \cdot \delta_0 + \frac{p}{2} \cdot \left(\delta_{1/\sqrt{p}} + \delta_{-1/\sqrt{p}}\right).$$

The denominator in (D.5) reduces to

$$(1-p) \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right) + \frac{p}{2} \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \left[\exp\left(-\frac{(y-\mu/\sqrt{p})^2}{2\sigma^2}\right) + \exp\left(-\frac{(y+\mu/\sqrt{p})^2}{2\sigma^2}\right)\right].$$

Moreover, it is easy to see that the enumerator of (D.5) reduces to

$$\frac{\sqrt{p}}{2} \cdot \left[\exp\left(-\frac{(y-\mu/\sqrt{p})^2}{2\sigma^2}\right) - \exp\left(-\frac{(y+\mu/\sqrt{p})^2}{2\sigma^2}\right)\right].$$

Numerical Denoising. For sparse Beta mixture (D.12), Gaussian mixture with aspect ratio (D.13) and sparse Gaussian mixture (D.10) distributions it is cumbersome to get a closed form expression for the optimal denoiser (6.18) in order to compute performance given a Haar design (6.16). We, thus, employ a typical binning in conjunction with Monte-Carlo to estimate the value of the conditional expectation in (6.18).

D.2 Experimental Details and Additional Numerical Results

D.2.1 Numerical Setup

Activation function and reparameterization of the weight matrix $\boldsymbol{B}$. Since the sign activation has derivative zero almost everywhere, it is not directly suited for gradient-based optimization. To overcome this issue for SGD training of the models described in the main body, we use a “straight-through” (see for example [YLZ+19]) approximation of it. In details, during the forward pass the activation of the network $\sigma(\cdot)$ is treated as a sign activation. However, during the backward pass (gradient computation) the derivatives are computed as if instead of $\sigma(\cdot)$ its relaxed version is used, namely, the tempered hyperbolic tangent:

$$\sigma_\tau(x) = \tanh\left(\frac{x}{\tau}\right).$$

We also note that such approximation is pointwise consistent except zero:

$$\lim_{\tau \to 0} = \sigma(x), \quad \forall\, \boldsymbol{x} \in \mathbb{R} \setminus \{0\}.$$

For the experiments we fix the temperature τ to the value of 0.1. Refining the approximation further, i.e., making τ smaller, does not affect the end result, but it makes numerics a bit less stable due to the increased variance of the derivative.

To ensure consistency of the "straight-though" approximation, we enforce the condition $\boldsymbol{B}_{i,:} \in \mathbb{S}^{d-1}$ via a simple differentiable reparameterization. Let $\boldsymbol{B} \in \mathbb{R}^{n \times d}$ be trainable network parameters, then

$$\hat{\boldsymbol{B}}_{i,:} = \frac{\boldsymbol{B}_{i,:}}{\|\boldsymbol{B}_{i,:}\|_2}.$$

It should be noted that it is not clear whether this constraint is necessary, since during the forward pass we use directly $\sigma(\cdot)$, which is agnostic to the row scaling of $\boldsymbol{B}$.

Augmentation and whitening. For the natural image experiments in Figures 6.5, 6.8 and D.8, we use data augmentation to bring the amount of images per class to the initial dataset scale. This step is crucial to simulate the minimization of the population risk and not the empirical one, when the number of samples per class is insufficient. We augment each image 15 times for CIFAR-10 data and 10 times for MNIST data. We note that the described amount of augmentation is sufficient: increasing it further does not change the results of the numerical experiments and only increases computational cost.

The whitening procedure corresponds to the matrix multiplication of each image by the inverse square root of the empirical covariance of the data. This is done to ensure that the data is isotropic (to be closer to the i.i.d. data assumption needed for the theoretical analysis). More formally, let $\boldsymbol{X} \in \mathbb{R}^{\mathrm{n}_{\mathrm{samples}} \times d}$ be the augmented data that is centered, i.e., the data mean is subtracted. Its empirical covariance is then given by

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{\mathrm{n}_{\mathrm{samples}} - 1} \cdot \sum_{i=1}^{\mathrm{n}_{\mathrm{samples}}} \boldsymbol{X}_{i,:} \boldsymbol{X}_{i,:}^\top.$$

In this view, the whitened data $\hat{\boldsymbol{X}} \in \mathbb{R}^{\mathrm{n}_{\mathrm{samples}} \times d}$ is obtained from the initial data $\boldsymbol{X}$ as follows

$$\hat{\boldsymbol{X}}_{i,:} = \hat{\boldsymbol{\Sigma}}^{-\frac{1}{2}} \boldsymbol{X}_{i,:},$$

where $\boldsymbol{X}_{i,:}$ defines the i-th data sample.

D.2.2 Phase Transition and Staircase in the Learning Dynamics for the Autoencoder in (6.2)

First, we provide an additional numerical simulation similar to the one in Figure 6.3 for the case of non-sparse Rademacher data, i.e., $p = 1$. Since condition (6.13) holds, we expect the minimizer to be a permutation of the identity, and the corresponding SGD dynamics to experience a staircase behaviour, as discussed in Section 6.4. Namely, the SGD algorithm first finds a random rotation that achieves Gaussian performance (indicated by the orange dashed line). Next, it searches a direction towards a sparse solution given by a permutation of the identity, and the corresponding loss remains at the plateau. Finally, the correct direction is found, and SGD quickly converges to the optimal solution.

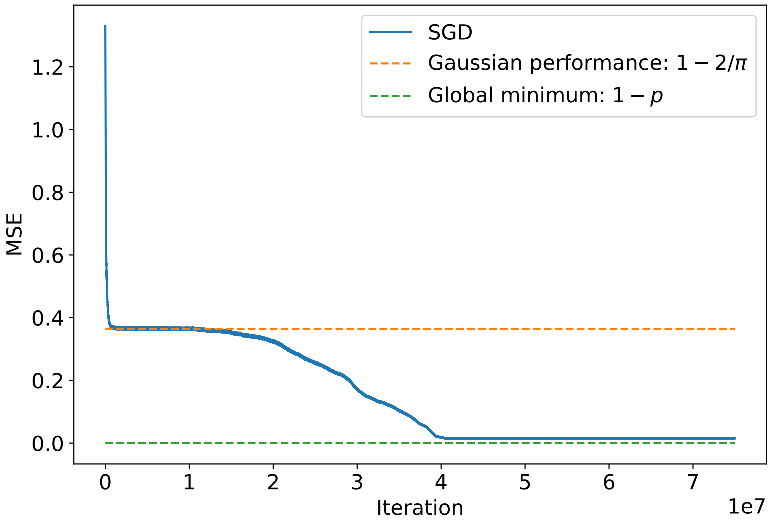


Figure D.1: Compression of Rademacher data ($p = 1$) via the autoencoder in (6.2). We set $d = 200$ and $r = 1$. The MSE is plotted as a function of the number of iterations, and it displays a staircase behavior.

Sparse Gaussian mixture. Next, we consider the compression of $\boldsymbol{x}$ with i.i.d. components distributed according to the following sparse mixture of Gaussians:

$$x_i \sim p \cdot \left(\frac{1}{2} \cdot \mathcal{N}\left(1, \frac{1-p}{p}\right) + \frac{1}{2} \cdot \mathcal{N}\left(-1, \frac{1-p}{p}\right)\right) + (1-p) \cdot \delta_0. \tag{D.10}$$

It is easy to verify that $\mathbb{E}[x_i^2] = 1$. In order to compute the transition point we need to access the first absolute moment of x_i, i.e., $\mathbb{E}|x_i|$. Using the result in [Win12], we are able to claim that

$$\mathbb{E}_{x\sim\mathcal{N}(\pm 1,\sigma^2)}|x| = \sigma\sqrt{\frac{2}{\pi}} \cdot \Phi\left(-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2\sigma^2}\right), \tag{D.11}$$

where $\Phi(a, b, c)$ stands for Kummer's confluent hypergeometric function:

$$\Phi(a, b, c) = \sum_{n=1}^{\infty} \frac{a^{\overline{n}}}{b^{\overline{n}}} \cdot \frac{c^n}{n!},$$

with $x^{\overline{n}}$ denoting the rising factorial, i.e.,

$$x^{\overline{n}} = z \cdot (z+1) \cdot \dots \cdot (z+n-1), \quad n \in \mathbb{N}_0.$$

We use `scipy.special.hyp1f1` to evaluate numerically $\Phi\left(-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2\sigma^2}\right)$, where $\sigma^2 = (1-p)/p$. Likewise, to find p_{crit} at which $\mathbb{E}|x_i| = \sqrt{\frac{2}{\pi}}$ we rely on numerics. The results are presented in Figure D.2.

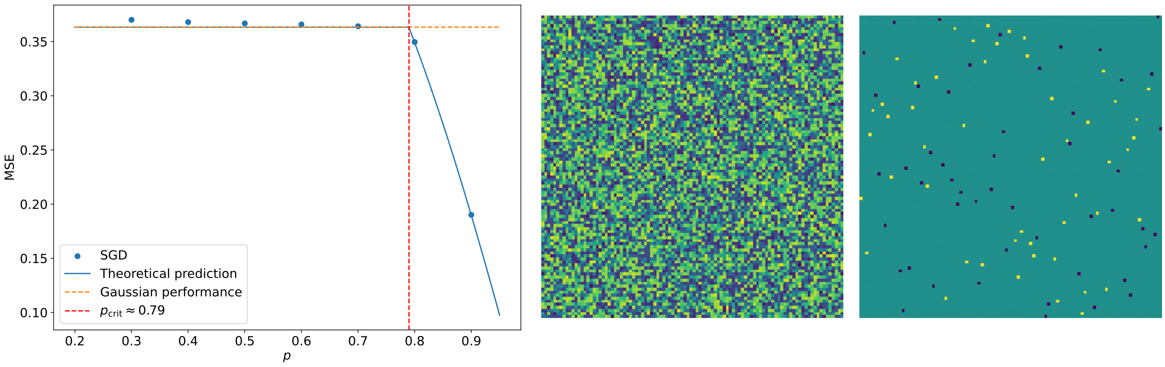


Figure D.2: Compression of data whose distribution is given by a sparse mixture of Gaussians via the autoencoder in (6.2). We set $d = 100$ and $r = 1$. *Left.* MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line): for $p < p_{\text{crit}}$, the loss is equal to the value obtained for Gaussian data, i.e., $1 - 2r/\pi$; for $p > p_{\text{crit}}$, the loss is smaller, and it is equal to $1 - r \cdot (\mathbb{E}|x_1|)^2$. *Center.* Encoder matrix $\boldsymbol{B}$ at convergence of SGD when $p = 0.6 < p_{\text{crit}}$: the matrix is a random rotation. *Right.* Encoder matrix $\boldsymbol{B}$ at convergence of SGD when $p = 0.9 \geq p_{\text{crit}}$. The negative sign in part of the entries of $\boldsymbol{B}$ is cancelled by the corresponding sign in the entries of $\boldsymbol{A}$. Hence, $\boldsymbol{B}$ is equivalent to a permutation of the identity.

We remark that the first absolute moment can always be estimated via Monte-Carlo sampling if a functional expression such as (D.11) is out of reach. We also note that the behaviour of the predicted curve after the transition point p_{crit} can be arbitrary. In particular, it is not always linear like in the case of sparse Rademacher data in Figure 6.2. For instance, in the case of the sparse Gaussian mixture of Figure D.2, the shape is clearly of non-linear nature.

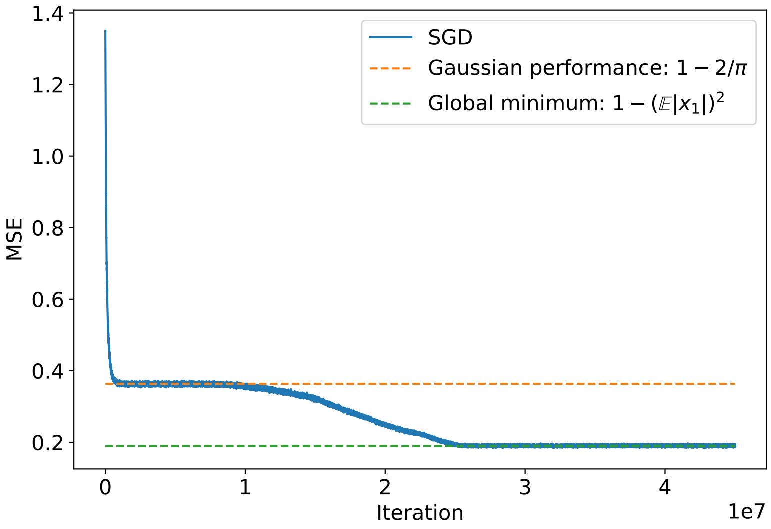


Figure D.3: Compression of data whose distribution is given by a sparse mixture of Gaussians via the autoencoder in (6.2). We set $d = 100$, $r = 1$, and $p = 0.9$. The MSE is plotted as a function of the number of iterations and, as $p > p_{\text{crit}}$, it displays a staircase behavior.

In Figure D.3, we provide an experiment similar to that of Figure 6.3, but for the compression of a sparse mixture of Gaussians with $p = 0.9$ at $r = 1$. We can clearly see that Figure D.3 again indicates the emergent staircase behaviour of the SGD loss for $p > p_{\text{crit}}$.

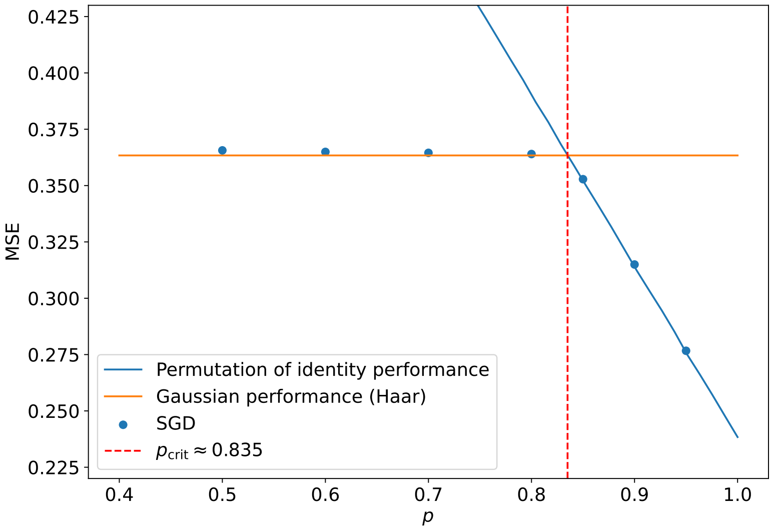


Figure D.4: Compression of data whose distribution is given by a sparse mixture of Beta distributions via the autoencoder in (6.2). We set $d = 100$ and $r = 1$. MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line): for $p < p_{\mathrm{crit}}$, the loss is equal to the value obtained for Gaussian data, i.e., $1 - 2r/\pi$; for $p > p_{\mathrm{crit}}$, the loss is smaller, and it is equal to $1 - r \cdot (\mathbb{E}|x_1|)^2$.

Sparse Beta mixture. Next, we consider the compression of $\boldsymbol{x}$ with i.i.d. components distributed according to the sparse mixture of Beta distributions with sparsity $(1 - p)$. The mixture is defined via the following sampling procedure:

$$\begin{aligned} &\hat{x}_i \sim \mathrm{Beta}(2, 5), \\ &\hat{x}_i \mapsto \mathrm{scale} \cdot \hat{m}_i \cdot \hat{x}_i, \quad \hat{m}_i \sim \mathrm{Rademacher}(0.5), \quad \text{(zero mean)}, \end{aligned}$$

where scale is such that $\mathrm{Var}(\hat{x}_i) = 1$. The final step of sampling is the addition of sparsity:

$$x_i = \frac{1}{\sqrt{p}} \cdot \hat{x}_i \cdot m_i, \quad m_i \sim \mathrm{Bernoulli}(p), \tag{D.12}$$

where $1/\sqrt{p}$ factor ensures $\mathrm{Var}(x_i) = 1$.

In this case, there is a phase transition at $p_{\mathrm{crit}} \approx 0.835$: for $p < p_{\mathrm{crit}}$, condition in (6.13) is not satisfied and GD converges to Haar weights giving the Gaussian MSE; for $p > p_{\mathrm{crit}}$, condition (15) is satisfied and GD converges to a sub-sampled permutation of the identity, which improves upon the Gaussian MSE. This is reported in the Figure D.4. To estimate the first absolute moment, i.e., $\mathbb{E}|x_1|$, for the corresponding permutation of identity performance plot, we use Monte-Carlo estimate over 10^7 samples.

Gaussian mixture with variable aspect ratio. Next, we consider the compression of $\boldsymbol{x}$ with i.i.d. components distributed according to the Gaussian mixture with varying aspect ratio γ. The mixture is defined via the following sampling procedure:

$$\begin{aligned} &\mu = 1, \quad \sigma = \mu \cdot \gamma \\ &x_i = \frac{1}{\sqrt{\mu^2 + \sigma^2}} \cdot m_i \cdot \hat{x}_i, \quad \hat{x}_i \sim \mathcal{N}(\mu, \sigma^2), \quad \hat{m}_i \sim \mathrm{Rademacher}(0.5). \end{aligned} \tag{D.13}$$

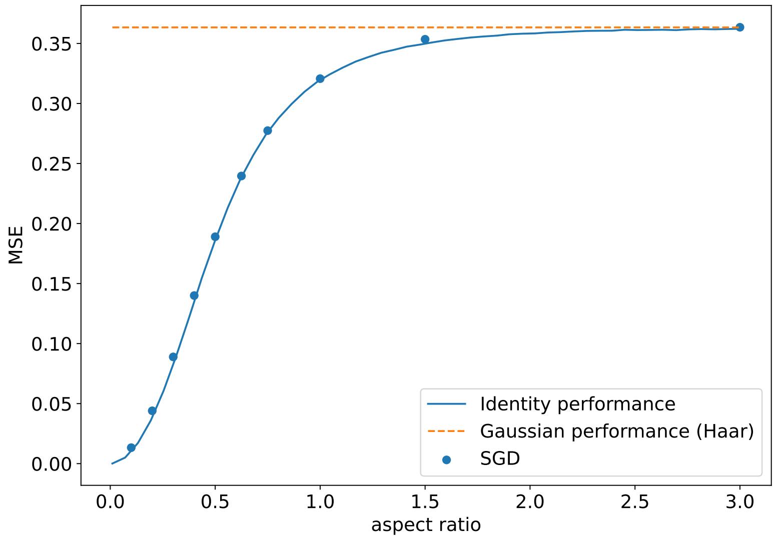


Figure D.5: Compression of data whose distribution is given by a (non-sparse) mixture of Gaussians via the autoencoder in (6.2). We set $d = 200$ and $r = 1$. MSE achieved by SGD at convergence, as a function of the aspect ratio γ. The empirical values (dots) match our theoretical prediction (blue line): $1 - r \cdot (\mathbb{E}|x_1|)^2$, which corresponds to the permutation of identity minimizer.

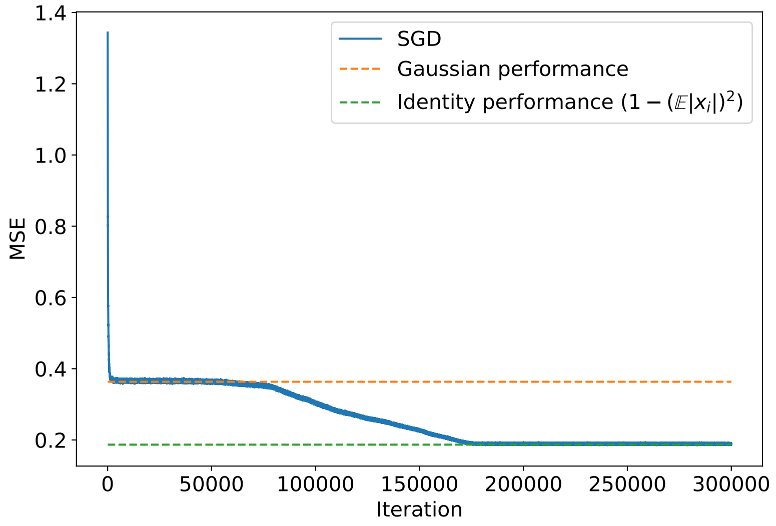


Figure D.6: Compression of data whose distribution is given by a (non-sparse) mixture of Gaussians via the autoencoder in (6.2). We set $d = 200$, $r = 1$, and aspect ratio $\gamma = 0.5$. The MSE is plotted as a function of the number of iterations and, as condition 6.13 is satisfied, it displays a staircase behavior.

Condition (6.13) is satisfied for all levels of γ and, as conjectured, SGD converges to a sub-sampled permutation of the identity, which improves upon the Gaussian MSE. This is reported in the Figure D.5.

Furthermore, the training loss exhibits a staircase behaviour: first the MSE rapidly converges to the Gaussian MSE (corresponding to Haar weights); then, there is a plateau; finally, the global minimum (corresponding to the permutation of identity weights) is reached. This is reported in Figure D.6.

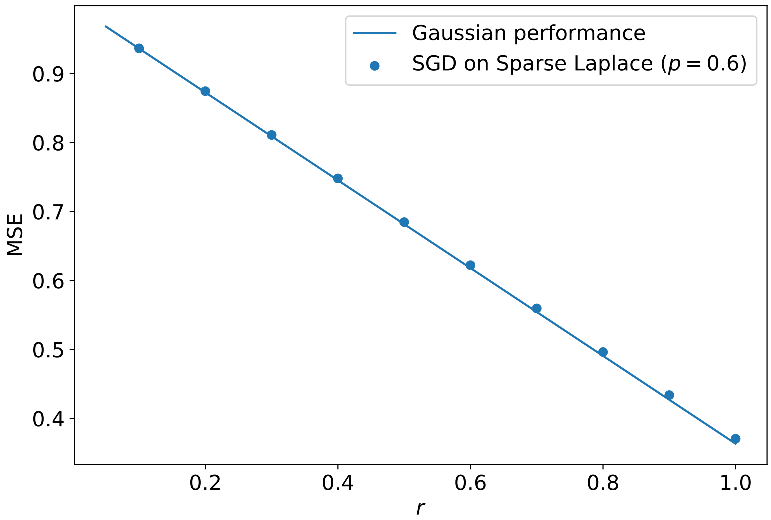


Figure D.7: Compression of data whose distribution is given by a sparse Laplace distribution via the autoencoder in (6.2). We set $d = 400$ and $p = 0.6$. The MSE is plotted as a function of the compression rate r and, as condition 6.13 is never satisfied, it displays Gaussian performace for all rates $r \leq 1$.

Sparse Laplace distribution. Next, we consider the compression of $\boldsymbol{x}$ with i.i.d. components distributed according to the sparse Laplace distribution (D.7). In this case, the condition (6.13) is never met regardless of the sparsity level p. In other words, the SGD will always converge to the Haar minimizer. We report the corresponding numerical values on Figure D.7 for different compression rates r, $d = 400$ and $p = 0.6$.

D.2.3 MNIST Experiment

In this subsection, we provide additional numerical evidence complementing the results presented in Figure 6.5. Namely, we provide a similar evaluation on Bernoulli-masked whitened MNIST data. As for the experiment in Figure 6.5, the sparsity level p is set to 0.7.

Note that the eigen-decomposition of the covariance of MNIST data has zero eigenvalues. In this case, we need to apply the lower bound from [SKHM23] that accounts for a degenerate spectrum. The corresponding result is stated in Theorem 5.2 of [SKHM23]. In particular, the number of zero eigenvalues n_0 is equal to 179, which means that at the value of the compression rate r given by

$$r = \frac{d - n_0}{d} = \frac{28^2 - 179}{28^2} \approx 0.77$$

the derivative of the lower bound experiences a jump-like behavior, as described in [SKHM23].

D.2.4 CIFAR-10: Laplace Approximation of Pixel Distribution

Figure D.9 demonstrates the quality of the Laplace approximation for whitened CIFAR-10 images. Namely, we note that the empirical distribution of the image pixels after whitening is well approximated by a Laplace random variable with unit second moment.

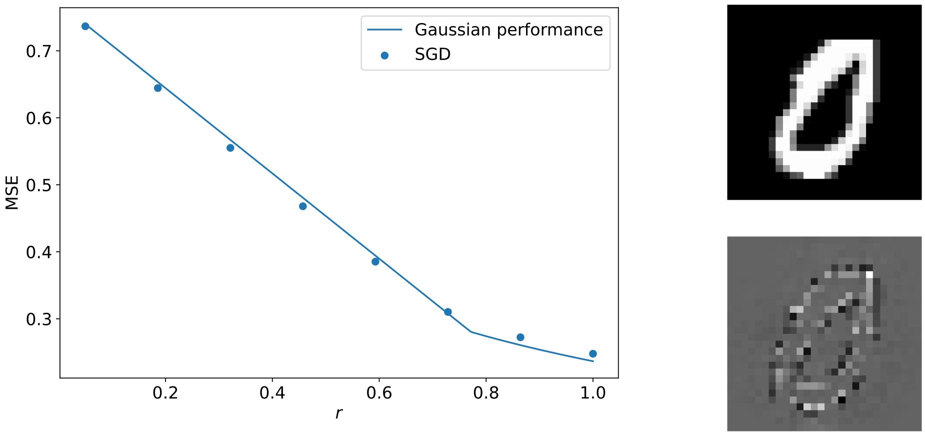


Figure D.8: Compression of masked and whitened MNIST images that correspond to digit "zero" via the two-layer autoencoder in (6.2). First, the data is whitened so that it has identity covariance (as in the setting of Theorem 9). Then, the data is masked by setting each pixel independently to 0 with probability $p = 0.7$. An example of an original image is on the top right, and the corresponding masked and whitened image is on the bottom right. The SGD loss at convergence (dots) matches the solid line, which corresponds to the prediction in (6.5) for the compression of standard Gaussian data (with no sparsity).
.

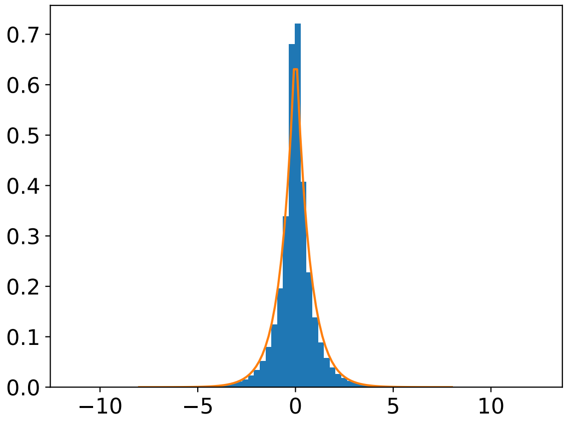


Figure D.9: Empirical distribution of whitened CIFAR-10 image pixels (blue histogram), and its approximation via a Laplace distribution with unit second moment (orange curve).

D.2.5 Provable Benefit of Nonlinearities for the Compression of Sparse Gaussian Data

Figure D.10 considers the compression of sparse Gaussian data, and it shows that the MSE achieved by the autoencoder in (6.4) with the optimal choice of f (namely, the RHS of (6.16) with $f = f^*$) is strictly lower than the MSE (6.5) achieved by the autoencoder in (6.2), for any sparsity level $p \in (0, 1)$. The conditional expectation $\mathbb{E}[x_1 | \mu x_1 + \sigma g]$ (cf. the definition of f^* in (6.18)) is computed numerically via a Monte-Carlo approximation.

D.2.6 Phase Transition and Staircase in the Learning Dynamics for the Autoencoder in (6.4)

Sparse Rademacher. For sparse Rademacher data, the optimal f^* given by (6.18) is computed explicitly in Appendix D.1.4 and plotted in Figure D.12. We note that functions

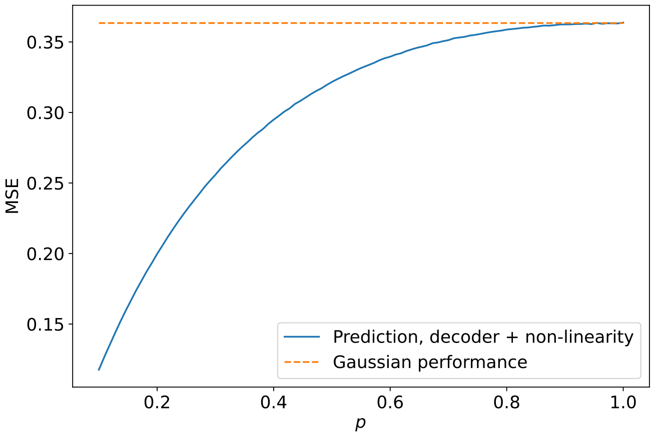


Figure D.10: Compression of sparse Gaussian data. We set $r = 1$. The solid blue line corresponds to the MSE in (6.16) with $f = f^*$ (defined in (6.18)), for different values of p; the dashed orange line corresponds to the Gaussian performance in (6.5), which is achieved by the autoencoder in (6.2).

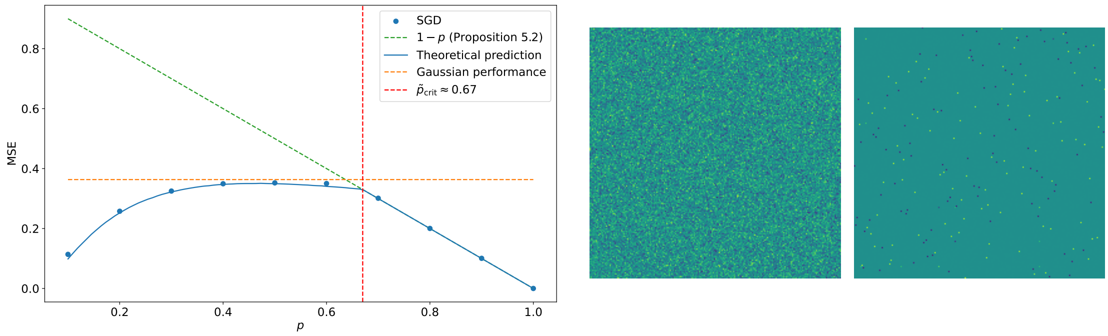


Figure D.11: Compression of sparse Rademacher data via the autoencoder in (6.4) with f of the form in (D.14). We set $d = 200$ and $r = 1$. *Left.* MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line). For $p < p_{\text{crit}}$, the loss is given by Proposition 6.5.1 for $\boldsymbol{B}$ sampled from the Haar distribution; for $p \geq p_{\text{crit}}$, the loss is given by Proposition 6.5.2 for $\boldsymbol{B}$ equal to the identity. *Center.* Encoder matrix $\boldsymbol{B}$ at convergence of SGD when $p = 0.3 < p_{\text{crit}}$: the matrix is a random rotation. *Right.* Encoder matrix $\boldsymbol{B}$ at convergence of SGD when $p = 0.7 \geq p_{\text{crit}}$. The negative sign in part of the entries of $\boldsymbol{B}$ is cancelled by the corresponding sign in the entries of $\boldsymbol{A}$. Hence, $\boldsymbol{B}$ is equivalent to a permutation of the identity.

of the form in (6.15) are unable to approximate f^* well. Thus, in the experiments we use a different parametric function for f given by the following mixture of hyperbolic tangents:

$$f(x) = \mathbb{1}_{x \geq 0} \cdot (\gamma_1 \cdot \tanh(\varepsilon_1 \cdot x - \alpha_1) + \beta_1) + \mathbb{1}_{x<0} \cdot (\gamma_2 \cdot \tanh(\varepsilon_2 \cdot x - \alpha_2) + \beta_2). \quad \text{(D.14)}$$

The numerical evaluation of the autoencoder in (6.4) with f of the form in (D.14) for the compression of sparse Rademacher data is provided in Figure D.11. We set $r = 1$ and $d = 200$. The solid blue line corresponds to the prediction of Proposition 6.5.1, obtained for random Haar $\boldsymbol{B}$; the solid orange line corresponds to the prediction of Proposition 6.5.2, obtained for $\boldsymbol{B}$ equal to the identity. The blue dots correspond to the performance of SGD, and they exhibit the transition in the learnt $\boldsymbol{B}$ from a random Haar matrix ($p < p_{\text{crit}}$) to a permutation of the identity ($p > p_{\text{crit}}$). The critical value p_{crit} is obtained from the intersection between the blue curve and the orange curve. For all values of p, the autoencoder in (6.4) outperforms

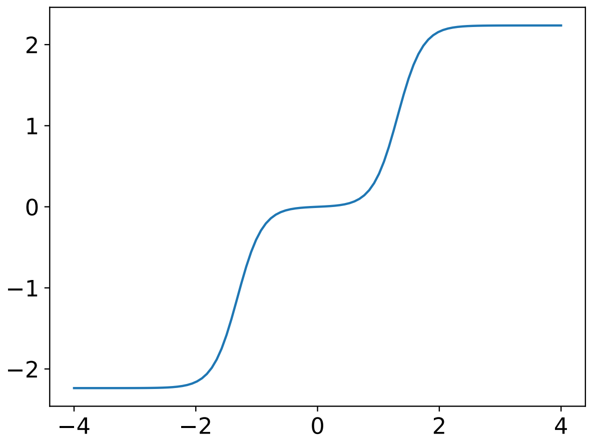


Figure D.12: Optimal f^* in (6.18) when x_1 is a sparse Rademacher random variable. We set $r = 1$ and $p = 0.2$.

the Gaussian MSE (6.5) (green dashed line) and, hence, it is able to exploit the structure in the data.

For $p > p_{\text{crit}}$, the staircase behavior of the SGD training dynamics is presented in Figure D.13.

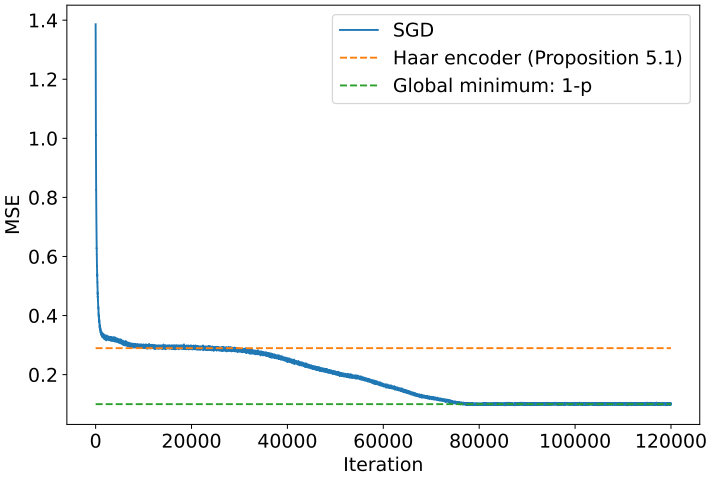


Figure D.13: Compression of sparse Rademacher data via the autoencoder in (6.4). We set $d = 200$, $r = 1$, and $p = 0.9$. The MSE is plotted as a function of the number of iterations and, as $p > p_{\text{crit}}$, it displays a staircase behavior.

Sparse Beta mixture. The numerical evaluation of the autoencoder in (6.4) for the data which comes from sparse Beta mixture (D.12) is illustrated on Figure D.14. As predicted by our theory, depending on the value of p, the optimal encoding corresponds either to a Haar design or a permutation of identity. The phase transition between two minimizers happens at the intersection of solid green and blue curves which correspond to the MSE of the respective minimizers. As discussed in Section D.1.4, in order to obtain values for the green curve, we use numerical estimate for the conditional expectation (6.18).

Sparse Gaussian mixture. The numerical evaluation of the autoencoder in (6.4) for the data which comes from sparse Gaussian mixture (D.10) is illustrated on Figure D.15. As predicted by our theory, depending on the value of p, the optimal encoding corresponds either to a Haar design or a permutation of identity. The phase transition between two minimizers

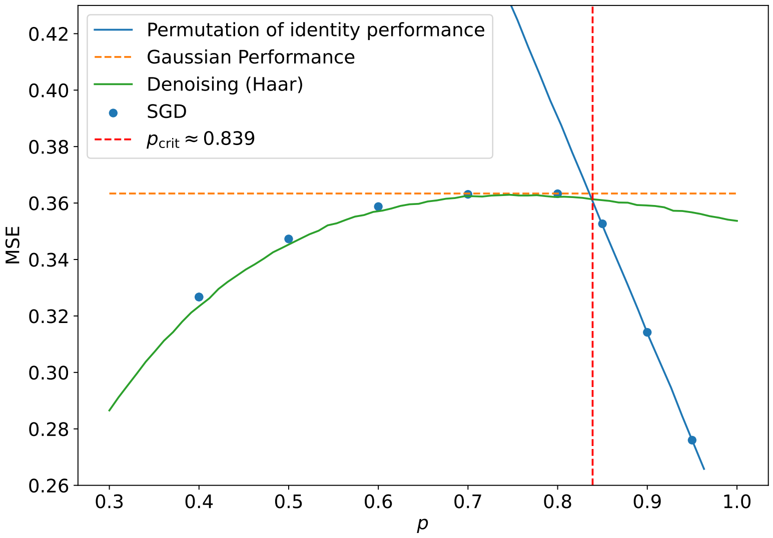


Figure D.14: Compression of data whose distribution is given by a sparse mixture of Beta distributions via the autoencoder in (6.2). We set $d = 200$ and $r = 1$. MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line). For $p < p_{\text{crit}}$, the loss is given by Proposition 6.5.1 for $\boldsymbol{B}$ sampled from the Haar distribution; for $p \geq p_{\text{crit}}$, the loss is given by Proposition 6.5.2 for $\boldsymbol{B}$ equal to the permutation of identity.

happens at the intersection of solid green and blue curves which correspond to the MSE of the respective minimizers. As discussed in Section D.1.4, in order to obtain values for the green curve, we use numerical estimate for the conditional expectation (6.18).

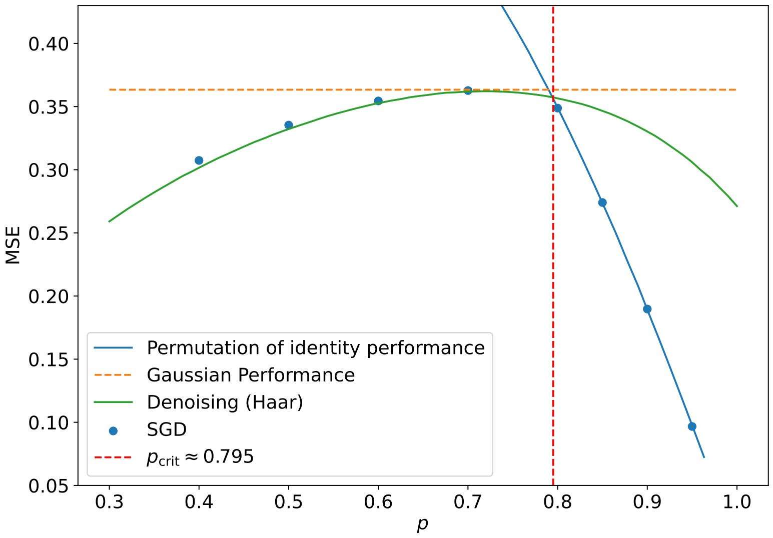


Figure D.15: Compression of data whose distribution is given by a sparse mixture of gaussians via the autoencoder in (6.2). We set $d = 200$ and $r = 1$. MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line). For $p < p_{\text{crit}}$, the loss is given by Proposition 6.5.1 for $\boldsymbol{B}$ sampled from the Haar distribution; for $p \geq p_{\text{crit}}$, the loss is given by Proposition 6.5.2 for $\boldsymbol{B}$ equal to the permutation of identity.

Gaussian mixture with aspect ratio. The numerical evaluation of the autoencoder in (6.4) for the data which comes from sparse Gaussian mixture (D.13) is illustrated on Figure D.16 for $d = 200$ and $r = 1$. As predicted by our theory, in this case, independently of aspect ratio value γ SGD converges to the minimizer which corresponds to a matrix $\boldsymbol{B}$ which is a permutation of identity, since the MSE value for Haar design 6.5.1 (dashed orange curve) is always inferior to the corresponding value achieved by permutation of identity 6.5.2 (solid blue curve). As discussed in Section D.1.4, in order to obtain values for the dashed orange curve, we use numerical estimate for the conditional expectation (6.18).

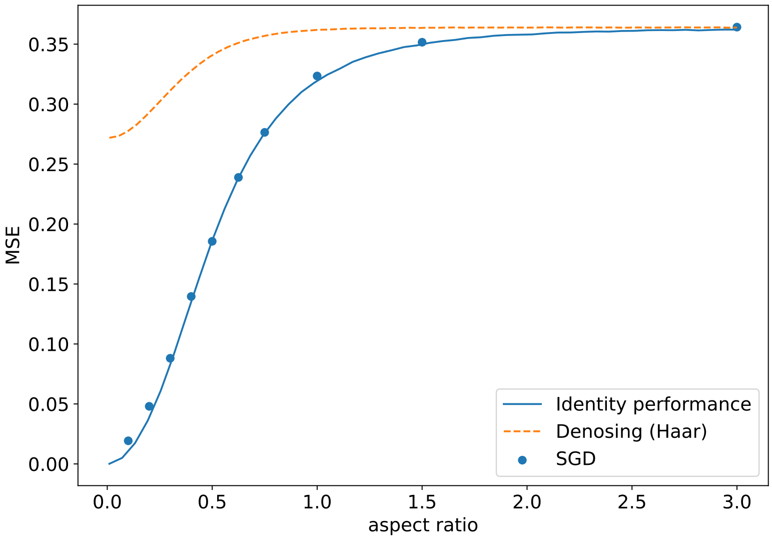


Figure D.16: Compression of data whose distribution is given by a (non-sparse) Gaussian mixture with aspect ratio via the autoencoder in (6.2). We set $d = 200$ and $r = 1$. MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (blue line) and the loss is given by Proposition 6.5.2 for $\boldsymbol{B}$ equal to a permutation of identity.

Sparse Laplace. The numerical evaluation of the autoencoder in (6.4) for the data which comes from sparse Laplace distribution (D.7) is illustrated on Figure D.17 for $d = 512$ and $r = 1$. As predicted by our theory, in this case, independently of sparsity level p, SGD converges to the minimizer which corresponds to an orthogonal matrix $\boldsymbol{B}$, since the MSE value for Haar design 6.5.1 (orange curve) is always superior to the corresponding value achieved by permutation of identity 6.5.2 (solid blue line). As discussed in Section D.1.4, in order to obtain values for the solid orange curve, we use numerical estimate for the conditional expectation (6.18).

D.2.7 Discussion on Multi-layer Decoder

First, let us elaborate on some design points for the network in (6.21). The merging operations $\oplus_2$ and $\oplus_3$ play the role of the correction terms $-\sum_{i=1}^{t-1} \beta_{t,i}\hat{\boldsymbol{x}}^i$ and $-\sum_{i=1}^{t} \alpha_{t,i}\hat{\boldsymbol{z}}^i$ in the RI-GAMP iterates in (6.20). Furthermore, the composition of $\oplus_3$ and $f_2(\cdot)$ in $\hat{\boldsymbol{x}}_2$ approximates taking the posterior mean in (6.20). We note that the network (6.21) can be generalized to emulate more RI-GAMP iterations, at the cost of additional layers and skip connections (induced by the merging operations $\oplus_k$).

In the rest of this appendix, we discuss how to obtain the orange curve in the right plot of Figure 6.9, which corresponds to the Bayes-optimal MSE when $\boldsymbol{B}$ is sampled from the Haar distribution. This optimal MSE is achieved by the fixed point of the VAMP algorithm proposed in [RSF19]. Thus, we implement the state evolution recursion from [RSF19], in

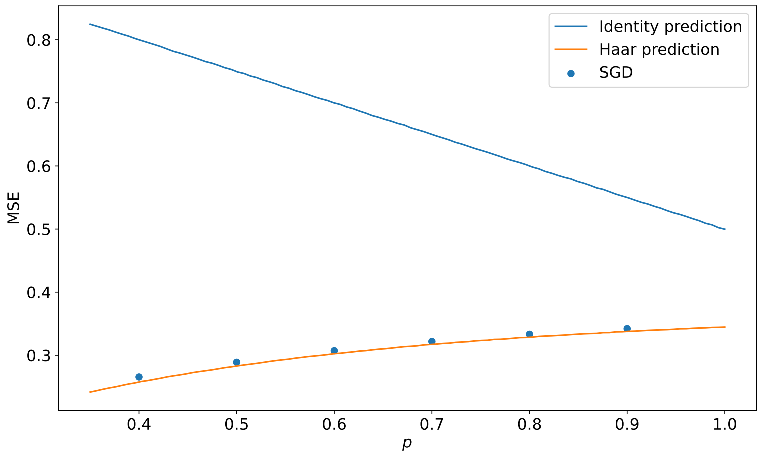


Figure D.17: Compression of data whose distribution is given by a sparse Laplace prior via the autoencoder in (6.2). We set $d = 512$ and $r = 1$. MSE achieved by SGD at convergence, as a function of the sparsity level p. The empirical values (dots) match our theoretical prediction (solid orange line) and the loss is given by Proposition 6.5.1 for $\boldsymbol{B}$ sampled from the Haar distribution.

order to evaluate the fixed point. As the specific setting considered here ($\boldsymbol{x} \sim \mathrm{SG}_d(p)$, $\boldsymbol{B}$ a Haar matrix, and a generalized linear model with sign activation) is not considered in [RSF19], we provide explicit expressions for the recursion leading to the desired MSE.

First state evolution function - $\mathcal{E}_1(\gamma_1)$. We start with the state evolution function that is equal to the following expected value of the derivative of the conditional expectation

$$\mathcal{E}_1(\gamma_1) = \mathbb{E}_{R_1}\left[\frac{\partial}{\partial R_1}\mathbb{E}[X|R_1 = X + P]\right], \quad X \sim \mathrm{SG}_1(p), \quad P \sim \mathcal{N}(0, \gamma_1^{-1}). \tag{D.15}$$

For completeness, we note that the quantity

$$\frac{\partial}{\partial R_1}\mathbb{E}[X|R_1 = X + P]$$

is in fact the conditional variance $\mathrm{Var}[X|R_1 = X + P]$ up to a scaling [DPS20], which is related to the optimal MSE.

Modulo the scalings, the computation of $\mathbb{E}[X|R_1 = X + P]$ is similar to the computation performed in Section D.1.4. For brevity, we just state the final result:

$$\begin{aligned}\mathbb{E}[X|R_1 = X + P] &= \frac{p \cdot \frac{R_1}{\sqrt{2\pi p^{-1}}} \cdot \exp\left(-\frac{pR_1^2}{2(p\gamma_1^{-1}+1)}\right) \cdot \frac{1}{(p\gamma_1^{-1}+1)^{3/2}}}{p \cdot \frac{1}{\sqrt{2\pi(p^{-1}+\gamma_1^{-1})}} \cdot \exp\left(-\frac{pR_1^2}{2(p\gamma_1^{-1}+1)}\right) + (1-p) \cdot \frac{1}{\sqrt{2\pi\gamma_1^{-1}}} \cdot \exp\left(-\frac{R_1^2}{2\gamma_1^{-1}}\right)} \\ &:= \frac{E(R_1)}{p(R_1)}.\end{aligned} \tag{D.16}$$

Taking the partial derivative in R_1 and substituting in (D.15) yields:

$$\begin{aligned}\mathcal{E}_1(\gamma_1) &= \gamma_1^{-1}\int_{\mathbb{R}} \frac{\partial}{\partial R_1}\mathbb{E}[X|R_1 = X + P]\cdot p(R_1)\mathrm{d}R_1 \\ &= \gamma_1^{-1}\int_{\mathbb{R}} \frac{E'(R_1)p(R_1) - E(R_1)p'(R_1)}{p^2(R_1)}p(R_1)\mathrm{d}R_1 \\ &= \gamma_1^{-1}\int_{\mathbb{R}} \left(E'(R_1) - E(R_1)\cdot\frac{\partial}{\partial R_1}\log p(R_1)\right)\mathrm{d}R_1.\end{aligned} \tag{D.17}$$

We can readily verify that

$$\int_{\mathbb{R}} E'(R_1)\mathrm{d}R_1 = \lim_{\text{ext}\to\infty} E(R_1)\Big|_{-\text{ext}}^{+\text{ext}} = 0.$$

An integration by parts for the remaining term in (D.17) gives:

$$\begin{aligned}\mathcal{E}_1(\gamma_1) = \gamma_1^{-1}\lim_{\text{ext}\to\infty} E(R_1)\log p(R_1)\Big|_{-\text{ext}}^{+\text{ext}} - \gamma_1^{-1}\int_{\mathbb{R}} E'(R_1)\log p(R_1)\mathrm{d}R_1 \\ = -\gamma_1^{-1}\int_{\mathbb{R}} E'(R_1)\log p(R_1)\mathrm{d}R_1.\end{aligned} \tag{D.18}$$

The RHS of (D.18) is then evaluated via numerical integration. For completeness, the derivative $E'(R_1)$ has the following form:

$$\begin{aligned}E'(R_1) &= p\cdot\frac{1}{\sqrt{2\pi p^{-1}}}\cdot\exp\left(-\frac{pR_1^2}{2(p\gamma_1^{-1}+1)}\right)\cdot\frac{1}{(p\gamma_1^{-1}+1)^{3/2}} \\ &\quad - p^2\cdot\frac{R_1^2}{\sqrt{2\pi p^{-1}}}\cdot\exp\left(-\frac{pR_1^2}{2(p\gamma_1^{-1}+1)}\right)\cdot\frac{1}{(p\gamma_1^{-1}+1)^{5/2}}.\end{aligned}$$

Second state evolution function - $\mathcal{E}_2(\tau_2, \gamma_2)$**.** This function is defined in terms of spectrum of $\boldsymbol{B}^\top\boldsymbol{B} \in \mathbb{R}^{d\times d}$. Namely, for $r \leq 1$, the distribution of the eigenvalues of $\boldsymbol{B}^\top\boldsymbol{B}$ obeys the following law

$$\rho_S = r\cdot\delta_1 + (1-r)\cdot\delta_0.$$

The state evolution function $\mathcal{E}_2(\tau_2, \gamma_2)$ is then defined as follows

$$\mathcal{E}_2(\tau_2,\gamma_2) := \mathbb{E}_{S\sim\rho_S}\left[\frac{1}{\tau_2\cdot S^2 + \gamma_2}\right] = r\cdot\frac{1}{\tau_2+\gamma_2} + (1-r)\cdot\frac{1}{\gamma_2}.$$

Third state evolution function - $\mathcal{B}_2(\tau_2, \gamma_2)$**.** The computation is similar to the case of the second state evolution function. Namely, the third state evolution function is defined as follows:

$$\mathcal{B}_2(\tau_2,\gamma_2) = \frac{1}{r}\cdot\mathbb{E}_{S\sim\rho_S}\left[\frac{\tau_2 S^2}{\tau_2 S^2+\gamma_2}\right] = \frac{1}{r}\cdot r\cdot\frac{\tau_2}{\tau_2+\gamma_2} = \frac{\tau_2}{\tau_2+\gamma_2}.$$

Fourth state evolution function - $\mathcal{B}_1(\tau_1)$**.** The last state evolution function is defined similarly to $\mathcal{E}_1(\gamma_1)$, namely

$$\mathcal{B}_1(\tau_1) = \mathbb{E}_{P_1,Y}\left[\frac{\partial}{\partial P_1}\mathbb{E}[Z|P_1, Y]\right]. \tag{D.19}$$

Here, $Z \sim \mathcal{N}(0, 1)$ has variance one (since the spectrum of $\boldsymbol{B}$ has unit variance), $Y = \mathrm{sign}(Z)$ and $P_1 = b \cdot Z + a \cdot G$, where $G \sim \mathcal{N}(0, 1)$ is independent of Z and

$$b = 1 - \tau_1^{-1}, \quad a = \sqrt{b \cdot (1 - b)}.$$

The outer expectation in (D.19) is estimated via Monte-Carlo. We now compute the conditional expectation. First note that the following decomposition (depending on the sign of Y) holds:

$$\mathbb{E}[Z|P_1, Y] = \mathbb{E}[Z'|P_1'], \tag{D.20}$$

where $Z' = \mathbb{1}_{ZY \geq 0} \cdot Z$ and $P_1' = b \cdot Z' + a \cdot G$. Using Bayes formula, we get that

$$\mathbb{E}[Z'|P_1'] = \frac{\int_{ZY \geq 0} Z \exp\left(-\frac{Z^2}{2}\right) \exp\left(-\frac{(P_1 - bZ)^2}{2a^2}\right) \mathrm{d}Z}{\int_{ZY \geq 0} \exp\left(-\frac{Z^2}{2}\right) \exp\left(-\frac{(P_1 - bZ)^2}{2a^2}\right) \mathrm{d}Z}. \tag{D.21}$$

Completing the square in the exponents gives

$$\frac{Z^2 a^2 + (P_1 - bZ)^2}{2a^2} = \frac{bZ^2 - 2bZP_1 + P_1^2}{2b(1-b)} = \frac{(Z - P_1)^2}{2(1-b)} + \frac{P_1^2}{2b},$$

which after substitution in (D.21) results in

$$\mathbb{E}[Z'|P_1'] = \frac{\int_{ZY \geq 0} Z \exp\left(-\frac{(Z-P_1)^2}{2\tau_1^{-1}}\right) \mathrm{d}Z}{\int_{ZY \geq 0} \exp\left(-\frac{(Z-P_1)^2}{2\tau_1^{-1}}\right) \mathrm{d}Z}. \tag{D.22}$$

Note that the denominator of (D.22) is easy to access via the standard Gaussian CDF $\Psi(\cdot)$ as follows

$$\begin{aligned} \frac{1}{\sqrt{2\pi\tau_1^{-1}}} \int_{ZY \geq 0} \exp\left(-\frac{(Z-P_1)^2}{2\tau_1^{-1}}\right) \mathrm{d}Z &= \mathbb{1}_{Y \geq 0} \cdot \left[1 - \Psi\left(-\frac{P_1}{\tau_1^{-1/2}}\right)\right] + \mathbb{1}_{Y<0} \cdot \Psi\left(-\frac{P_1}{\tau_1^{-1/2}}\right) \\ &= \Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right), \end{aligned} \tag{D.23}$$

where for the last equality we use that $\Psi(x) = 1 - \Psi(-x)$ and $Y \in \{-1, +1\}$. For the numerator of (D.22), we get

$$\frac{1}{\sqrt{2\pi\tau_1^{-1}}} \int \mathbb{1}_{YZ \geq 0} \cdot Z \exp\left(-\frac{(Z-P_1)^2}{2\tau_1^{-1}}\right) \mathrm{d}Z. \tag{D.24}$$

Let us denote the PDF of $\mathcal{N}(\mu, \sigma^2)$ by ρ_{μ,σ^2}, and use the shorthand $\rho(\cdot)$ for $\rho_{0,1}(\cdot)$. Note that $\rho_{x,\sigma^2}(0) = \sigma^{-1}\rho(x/\sigma)$. Then, by Stein's identity, we have

$$\mathbb{E}\left[\mathbb{1}_{YZ \geq 0} \cdot (Z - P_1)\right] = \tau_1^{-1} \cdot \mathbb{E}[Y \cdot \delta_0(Z)] = Y\tau_1^{-1} \cdot \rho_{P_1, \tau_1^{-1}}(0) = Y\tau_1^{-1/2} \cdot \rho\left(\frac{P_1}{\tau_1^{-1/2}}\right),$$

as the weak derivative of $\mathbb{1}_{YZ \geq 0}$ is well-defined and equal to $Y \cdot \delta_0(Z)$. Noting that similarly to (D.23)

$$\mathbb{E}\left[\mathbb{1}_{YZ \geq 0} \cdot P_1\right] = P_1 \cdot \Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right),$$

we conclude that

$$\text{(D.24)} = P_1 \cdot \Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right) + Y\tau_1^{-1/2} \cdot \rho\left(\frac{P_1}{\tau_1^{-1/2}}\right). \tag{D.25}$$

Combining the results gives

$$\mathbb{E}[Z_1'|P_1'] = \frac{P_1 \cdot \Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right) + Y\tau_1^{-1/2} \cdot \rho\left(\frac{P_1}{\tau_1^{-1/2}}\right)}{\Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right)} = P_1 + Y\tau_1^{-1/2} \cdot \frac{\rho\left(\frac{P_1}{\tau_1^{-1/2}}\right)}{\Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right)}. \tag{D.26}$$

It now remains to take the derivative in P_1. We get that

$$\mathcal{B}_1(\tau_1) = 1 - \frac{YP_1\sqrt{\tau_1} \cdot \rho\left(\frac{P_1}{\tau_1^{-1/2}}\right) \cdot \Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right) + \rho\left(\frac{P_1}{\tau_1^{-1/2}}\right)^2}{\Psi\left(\frac{YP_1}{\tau_1^{-1/2}}\right)^2}, \tag{D.27}$$

where we used that $Y^2 = 1$ and that $\frac{\partial}{\partial x}\Psi(x) = \rho(x)$.

State evolution recursion. At this point, we are ready to present the state evolution recursion, which reads

$$\begin{aligned}
\gamma_{2,k} &= \gamma_{1,k} \cdot \frac{1 - \mathcal{E}_1(\gamma_{1,k})}{\mathcal{E}_1(\gamma_{1,k})}, \\
\tau_{2,k} &= \tau_{1,k} \cdot \frac{1 - \mathcal{B}_1(\tau_{1,k})}{\mathcal{B}_1(\tau_{1,k})}, \\
\gamma_{1,k+1} &= \gamma_{2,k} \cdot \frac{1 - \mathcal{E}_2(\tau_{2,k}, \gamma_{2,k})}{\mathcal{E}_2(\tau_{2,k}, \gamma_{2,k})} = \gamma_{2,k} \cdot \frac{r \cdot \tau_{2,k}}{(1-r) \cdot \tau_{2,k} + \gamma_{2,k}}, \\
\tau_{1,k+1} &= \tau_{2,k} \cdot \frac{1 - \mathcal{B}_2(\tau_{2,k}, \gamma_{2,k})}{\mathcal{B}_2(\tau_{2,k}, \gamma_{2,k})} = \gamma_{2,k}.
\end{aligned} \tag{D.28}$$

The initialization $\gamma_{1,0}$ and $\tau_{1,0}$ can be set to a small strictly positive number. For the experiments, we choose the value of 10^{-6}.

MSE from the state evolution parameter $\gamma_{1,k+1}$. The MSE after k steps of the recursion can be accessed via the function previously computed in (D.16). Namely, let $\boldsymbol{x} \sim \mathrm{SG}_d(p)$ and $\boldsymbol{r}_1 = \boldsymbol{x} + \boldsymbol{p}$, where $\boldsymbol{p}$ has i.i.d. entries with distribution $\mathcal{N}(0, \gamma_{1,k+1}^{-1})$. Define

$$g(\boldsymbol{r}_1) = \mathbb{E}[\boldsymbol{x}|\boldsymbol{r}_1 = \boldsymbol{x} + \boldsymbol{p}].$$

By the tower property of the conditional expectation, we claim that the following holds

$$\mathbb{E}[\mathbb{E}[X|Y] \cdot X] = \mathbb{E}[\mathbb{E}[\mathbb{E}[X|Y] \cdot X|Y]] = \mathbb{E}\left[(\mathbb{E}[X|Y])^2\right],$$

where we use that $\mathbb{E}[X|Y]$ is measurable w.r.t. Y. Thus, we have that

$$\mathbb{E}\langle g(\boldsymbol{r}_1), \boldsymbol{x}\rangle = d \cdot \mathbb{E}\left[(g(\boldsymbol{r}_1)_1)^2\right],$$

where $g(\boldsymbol{r}_1)_1$ denotes the first entry of the vector $g(\boldsymbol{r}_1)$. Finally, the desired MSE after k steps of the recursion is equal to

$$d^{-1} \cdot \mathbb{E}\|\boldsymbol{x} - g(\boldsymbol{r}_1)\|_2^2 = 1 - \mathbb{E}\left[(g(\boldsymbol{r}_1)_1)^2\right]. \tag{D.29}$$

We evaluate (D.29) for k large enough, so that the MSE has converged. For the experiment in Figure 6.9, we use $k = 15$, as for $k \geq 15$ the MSE value in (D.29) is stable.