École Polytechnique Fédérale de Lausanne

Doctoral Program in Physics (EDPY)

Effective Dynamics of Stochastic Gradient Descent in High-Dimensional Neural Networks

---

to be presented on 26th August 2026

from

Luca Arnaboldi

Composition du jury :

Directeur de thèse	Prof. Florent Krzakala (EPFL)
Jury interne	Prof. Nicolas Flammarion (EPFL)
Jury externes	Prof. Sebastian Goldt (SISSA) Prof. Joan Bruna (NYU)

Lausanne, 2026

"Roads? Where we're going, we don't need roads."

— Emmett L. Brown, Ph.D.

Abstract

Understanding why gradient-trained neural networks succeed requires theoretical models that capture how they discover low-dimensional structure buried in high-dimensional data. Despite the empirical dominance of large-scale deep learning, even small, shallow networks remain analytically elusive: the dynamics that govern what they learn, and how fast, are not yet fully understood. This thesis builds on a statistical-physics-inspired theory of these dynamics, studying stochastic gradient descent on shallow networks and simplified attention architectures trained on synthetic, teacher-generated data, by tracking the evolution of a handful of summary statistics rather than every individual weight. In the high-dimensional limit, this evolution is described exactly by a closed system of deterministic differential equations, making the speed and difficulty of learning fully quantifiable.

Initially, we frame previously separate descriptions of these dynamics — the classical gradient-flow, high-dimensional, and overparameterized mean-field regimes — as special cases of a single underlying process. Within this framework, we give closed-form sample-complexity formulas for learning hard, phase-retrieval-type targets, and show that increasing network width speeds up learning only by a constant factor, without escaping the hardness controlled by the target's information exponent.

The second part studies how simple modifications to vanilla stochastic gradient descent move training out of this hardness landscape. A small amount of data reuse, seeing each sample twice rather than once, shifts the obstruction away from the *information exponent* that governs one-pass learning, to a smaller *generative exponent*, recovering almost any hidden direction efficiently, with a hierarchical mechanism accounting for the remaining hard cases. We identify the batch size that minimizes training time for a target of given difficulty, and introduce a modified loss that removes the resulting speed limit altogether.

Finally, we extend the analysis beyond static inputs to sequential data and a simplified attention layer, showing that positional structure can substantially accelerate learning relative to unstructured inputs.

Together, these results give a solvable, unified theory of the training dynamics of shallow networks and simple attention models. They quantify when gradient-based learning succeeds at uncovering hidden structure, how far simple algorithmic changes can push that success, and which properties of the data and training govern its speed.

Keywords: deep learning, neural networks, stochastic gradient descent, training dynamics, statistical physics, mean-field theory, teacher-student models, sample complexity, phase retrieval, attention mechanisms

Sommario

Per comprendere il motivo per cui le reti neurali addestrate con il metodo del gradiente ottengono risultati positivi, sono necessari modelli teorici in grado di cogliere il modo in cui esse individuano strutture a bassa dimensionalità nascoste all'interno di dati ad alta dimensionalità. Nonostante il predominio empirico del deep learning su larga scala, anche le reti di piccole dimensioni e poco profonde rimangono analiticamente sfuggenti: le dinamiche che regolano ciò che apprendono, e la velocità con cui lo fanno, non sono ancora pienamente comprese. La presente tesi si basa su una teoria di tali dinamiche ispirata alla fisica statistica: studia la discesa stocastica del gradiente su reti poco profonde e su architetture di attenzione semplificate, addestrate su dati sintetici generati da un «insegnante», monitorando l'evoluzione di un ristretto numero di statistiche riassuntive anziché di ogni singolo peso. Nel limite ad alta dimensionalità, tale evoluzione è descritta esattamente da un sistema chiuso di equazioni differenziali deterministiche, rendendo la velocità e la difficoltà dell'apprendimento pienamente quantificabili.

Inizialmente, inquadriamo descrizioni di questa dinamica precedentemente separate — i regimi classici del flusso di gradiente, ad alta dimensionalità e del campo medio sovraparametrizzato — come casi particolari di un unico processo sottostante. All'interno di questo quadro, forniamo formule in forma chiusa relative alla complessità campionaria per l'apprendimento di obiettivi difficili, del tipo «recupero di fase» (phase retrieval), e dimostriamo che l'aumento della larghezza della rete accelera l'apprendimento solo di un fattore costante, senza sfuggire alla difficoltà determinata dall'esponente di informazione dell'obiettivo.

La seconda parte studia come semplici modifiche alla discesa stocastica del gradiente standard (vanilla stochastic gradient descent) consentano di spostare l'addestramento al di fuori di questo panorama di difficoltà. Un modesto riutilizzo dei dati, ovvero l'osservazione di ciascun campione due volte anziché una sola, sposta l'ostacolo dall'*esponente di informazione* (information exponent) che governa l'apprendimento a passaggio singolo verso un *esponente generativo* (generative exponent) più piccolo, consentendo di recuperare in modo efficiente quasi tutte le direzioni nascoste, con un meccanismo gerarchico che tiene conto dei casi difficili rimanenti. Identifichiamo la dimensione del batch che minimizza il tempo di addestramento per un obiettivo di data difficoltà e introduciamo una funzione di perdita modificata che elimina del tutto il limite di velocità risultante.

Infine, estendiamo l'analisi oltre gli input statici ai dati sequenziali e a un livello di attenzione semplificato, dimostrando che la struttura posizionale può accelerare sostanzialmente l'apprendimento rispetto agli input non strutturati.

Nel loro insieme, questi risultati forniscono una teoria risolvibile e unificata delle dinamiche di

addestramento delle reti poco profonde e dei modelli di attenzione semplici. Essi quantificano quando l'apprendimento basato sui gradienti riesce a scoprire la struttura nascosta, fino a che punto semplici modifiche algoritmiche possano potenziare tale successo e quali proprietà dei dati e dell'addestramento ne determinino la velocità.

Parole chiave: apprendimento profondo, reti neurali, discesa del gradiente stocastica, dinamiche di addestramento, fisica statistica, teoria di campo medio, modelli insegnante-studente (teacher-student), complessità campionaria, recupero di fase, meccanismi di attenzione

Résumé

Pour comprendre pourquoi les réseaux neuronaux entraînés à l'aide de la méthode du gradient obtiennent des résultats positifs, nous avons besoin de modèles théoriques capables de rendre compte de la manière dont ils identifient des structures cachées de faible dimension au sein de données de haute dimension. Malgré la prédominance empirique de l'apprentissage profond à grande échelle, même les petits réseaux peu profonds restent difficiles à appréhender d'un point de vue analytique : les dynamiques qui régissent ce qu'ils apprennent — ainsi que la vitesse à laquelle ils le font — ne sont pas encore pleinement comprises. Cette thèse s'appuie sur une théorie de ces dynamiques inspirée de la physique statistique : elle étudie la descente stochastique du gradient sur des réseaux peu profonds et des architectures d'attention simplifiées, entraînés sur des données synthétiques générées par un « enseignant », en suivant l'évolution d'un petit nombre de statistiques récapitulatives plutôt que celle de chaque poids individuel. Dans la limite des dimensions élevées, cette évolution est décrite exactement par un système fermé d'équations différentielles déterministes, ce qui rend la vitesse et la difficulté de l'apprentissage entièrement quantifiables.

Dans un premier temps, nous replaçons les descriptions auparavant distinctes de cette dynamique — les régimes classiques du flux de gradient, de la haute dimensionnalité et du champ moyen surparamétré — dans le cadre de cas particuliers d'un processus sous-jacent unique. Dans ce cadre, nous fournissons des expressions sous forme close pour la complexité d'échantillonnage de l'apprentissage d'objectifs difficiles, tels que la « récupération de phase », et démontrons que l'augmentation de la largeur du réseau n'accélère l'apprentissage que d'un facteur constant, sans échapper à la difficulté déterminée par l'exposant d'information de l'objectif.

La deuxième partie examine comment de simples modifications apportées à la méthode standard (classique) de descente stochastique du gradient permettent de déplacer l'entraînement hors de ce paysage de difficulté. Une réutilisation modérée des données — c'est-à-dire observer chaque échantillon deux fois au lieu d'une seule — déplace l'obstacle de l'*exposant d'information* (information exponent) régissant l'apprentissage en un seul passage vers un *exposant génératif* (generative exponent) plus faible, permettant ainsi la récupération efficace de la quasi-totalité des directions cachées, grâce à un mécanisme hiérarchique qui prend en compte les cas difficiles restants. Nous identifions la taille de lot qui minimise le temps d'entraînement pour une cible de difficulté donnée et introduisons une fonction de perte modifiée qui élimine complètement la limite de vitesse qui en résulte.

Enfin, nous étendons l'analyse au-delà des entrées statiques aux données séquentielles et à une couche d'attention simplifiée, démontrant que la structure positionnelle peut accélérer considérablement l'apprentissage par rapport à des entrées non structurées.

Pris dans leur ensemble, ces résultats fournissent une théorie unifiée et résolvable de la dynamique d'entraînement des réseaux peu profonds et des modèles d'attention simples. Ils quantifient à quel moment l'apprentissage basé sur les gradients parvient à découvrir une structure cachée, dans quelle mesure de simples modifications algorithmiques peuvent améliorer ce résultat, et quelles propriétés des données et de l'entraînement déterminent sa vitesse.

Mots-clés : apprentissage profond, réseaux de neurones, descente de gradient stochastique, dynamiques d'entraînement, physique statistique, théorie du champ moyen, modèles enseignant-élève (teacher-student), complexité d'échantillonnage, récupération de phase, mécanismes d'attention

Acknowledgements

To be written after private defense.

Foreword

In this thesis, we study the training dynamics of neural networks under stochastic gradient descent, with a focus on characterizing *when* and *how fast* learning happens. Our analysis is inspired by the statistical physics of high-dimensional systems: we work in the limit where the input dimension is large and the number of samples needed to learn grows with it at a rate set by the difficulty of the target, and we use low-dimensional equations that describe the full training trajectory. The central organizing concept is the *information exponent*, an integer that measures the flatness of the loss landscape near initialization and sets the sample complexity of one-pass SGD. It is our point of departure rather than the final word: much of what follows is devoted to charting where this baseline breaks down, showing how a change of training protocol or of architecture replaces the information exponent with a related quantity—a generative, leap, or sequence exponent—that governs learning in the new setting.

To set the stage, Chapter 1 develops the mathematical framework that underlies the subsequent chapters. We motivate the statistical physics approach, introduce the teacher–student setup that serves as our laboratory throughout the thesis, and derive the Saad–Solla ordinary differential equations governing online SGD in two-layer networks. We then show how the information exponent emerges from these equations as the key quantity controlling escape from the initialization saddle, and we discuss its generalizations to multi-index functions and broader algorithm classes.

Chapter 2 provides a unifying derivation of dimensionless ODEs for SGD, reconciling the high-dimensional and mean-field viewpoints within a single framework and identifying the minimal set of order parameters needed to describe training. Chapter 3 applies this framework to characterize the escape from mediocre initialization when learning hard single-index models such as phase retrieval, giving sharp conditions on the learning rate and noise level under which SGD escapes mediocrity or stalls.

Chapters 4 and 5 extend the theory in two complementary directions. Chapter 4 examines the role of batch size, showing that it governs a fundamental time–complexity tradeoff: enlarging the batch up to a critical size, which grows with the hardness of the target, proportionally reduces the number of gradient steps needed to learn while leaving the total number of samples consumed unchanged, so the gain is one of parallelism rather than of sample efficiency. Chapter 5 turns to multi-index models, where the target function depends on several latent directions simultaneously, and shows that a simple data-repetition scheme restores learnability in this harder setting: revisiting the same data lets the network sidestep the barrier set by the information exponent and

learn almost every direction in a number of steps nearly linear in the input dimension.

Chapter 6 extends the framework beyond fully connected two-layer networks to sequence models and single-layer attention networks. We derive the SGD asymptotics in this architecturally richer setting and show that analogous effective equations govern learning, with a *sequence* information exponent taking over the organizing role once the architecture introduces structured dependencies between tokens.

Each chapter is typically accompanied by its own repository containing the code to reproduce its numerical experiments; see the corresponding paper's introduction for the link. The code for the additional numerical experiments added specifically for this thesis and not present in the original papers, together with the full LaTeX source of the manuscript and the Claude Code Skills used to assemble it, is available at `https://github.com/arn4/phd-thesis`.

Publications

This thesis is a collection of 5 papers I have co-authored during my PhD, carefully reviewed and edited to be included in the thesis. Below I summarize each paper's contribution.

1. From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layer networks

Luca Arnaboldi, Ludovic Stephan, Florent Krzakala, Bruno Loureiro
Conference on Learning Theory (COLT) 2023

Abstract. This manuscript investigates the one-pass stochastic gradient descent (SGD) dynamics of a two-layer neural network trained on Gaussian data and labels generated by a similar, though not necessarily identical, target function. We rigorously analyse the limiting dynamics via a deterministic and low-dimensional description in terms of the sufficient statistics for the population risk. Our unifying analysis bridges different regimes of interest, such as the classical gradient-flow regime of vanishing learning rate, the high-dimensional regime of large input dimension, and the overparameterised "mean-field" regime of large network width, covering as well the intermediate regimes where the limiting dynamics is determined by the interplay between these behaviours. In particular, in the high-dimensional limit, the infinite-width dynamics is found to remain close to a low-dimensional subspace spanned by the target principal directions. Our results therefore provide a unifying picture of the limiting SGD dynamics with synthetic data.

Contribution. I have contributed all numerical experiments and illustrations. I have helped design part of the theoretical framework.

2. Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD

Luca Arnaboldi, Florent Krzakala, Bruno Loureiro, Ludovic Stephan
arXiv preprint 2023, accepted at NeurIPS 2023 Workshop OPT

Abstract. This study explores the sample complexity for two-layer neural networks to learn a generalized linear target function under Stochastic Gradient Descent (SGD), focusing on the challenging regime where many flat directions are present at initialization. It is well-established that in this scenario $n = O(d \log d)$ samples are typically needed. However, we provide precise results concerning the pre-factors in high-dimensional contexts and for varying widths. Notably, our findings suggest that

overparameterization can only enhance convergence by a constant factor within this problem class. These insights are grounded in the reduction of SGD dynamics to a stochastic process in lower dimensions, where escaping mediocrity equates to calculating an exit time. Yet, we demonstrate that a deterministic approximation of this process adequately represents the escape time, implying that the role of stochasticity may be minimal in this scenario.

CONTRIBUTION. I have authored the theoretical framework and the numerical experiments.

3. Online Learning and Information Exponents: On The Importance of Batch size, and Time / Complexity Tradeoffs

Luca Arnaboldi, Yatin Dandi, Florent Krzakala, Bruno Loureiro, Luca Pesce, Ludovic Stephan
International Conference on Machine Learning (ICML) 2024

ABSTRACT. We study the impact of the batch size n_b on the iteration time T of training two-layer neural networks with one-pass stochastic gradient descent (SGD) on multi-index target functions of isotropic covariates. We characterize the optimal batch size minimizing the iteration time as a function of the hardness of the target, as characterized by the information exponents. We show that performing gradient updates with large batches $n_b \lesssim d^{\ell/2}$ minimizes the training time without changing the total sample complexity, where ℓ is the information exponent of the target to be learned [BGJ21] and d is the input dimension. However, larger batch sizes than $n_b \gg d^{\ell/2}$ are detrimental for improving the time complexity of SGD. We provably overcome this fundamental limitation via a different training protocol, *Correlation loss SGD*, which suppresses the self-interaction term in the loss function. We show that one can track the training progress by a system of low-dimensional ordinary differential equations (ODEs). Finally, we validate our theoretical results with numerical experiments.

CONTRIBUTION. I have contributed to the problem formulation and design. I have designed the code for the numerical experiments, and executed a portion of them.

4. Repetita Iuvant: Data Repetition Allows SGD to Learn High-Dimensional Multi-Index Functions

Luca Arnaboldi, Yatin Dandi, Florent Krzakala, Luca Pesce, Ludovic Stephan
arXiv preprint 2024

ABSTRACT. Neural networks can identify low-dimensional relevant structures within high-dimensional noisy data, yet our mathematical understanding of how they do so remains scarce. Here, we investigate the training dynamics of two-layer shallow neural networks trained with gradient-based algorithms, and discuss how they learn pertinent features in multi-index models, that is target functions with low-dimensional relevant directions. In the high-dimensional regime, where the input dimension d diverges, we show that a simple modification of the idealized single-pass gradient descent training scenario, where data can now be repeated or iterated upon twice, drastically improves its computational efficiency. In particular, it surpasses the limitations previously believed to be dictated by the Information and Leap exponents associated with the target function to be learned. Our results highlight the ability of networks to learn relevant structures from data alone without any pre-processing. More precisely, we show that (almost) all directions are learned with at most $O(d \log d)$ steps. Among the exceptions is a set of hard functions that includes sparse parities. In the presence of coupling between directions, however, these can be learned sequentially through a hierarchical mechanism that generalizes the notion of staircase functions. Our results are proven by a rigorous study of the evolution of the relevant statistics for high-dimensional dynamics.

CONTRIBUTION. I have contributed to the problem formulation and design. I have designed the code for the numerical experiments, and executed a portion of them.

5. Asymptotics of SGD in Sequence-Single Index Models and Single-Layer Attention Networks

Luca Arnaboldi, Bruno Loureiro, Ludovic Stephan, Florent Krzakala, Lenka Zdeborová

Neural Information Processing Systems (NeurIPS) 2025

ABSTRACT. We study the dynamics of stochastic gradient descent (SGD) for a class of sequence models termed Sequence Single-Index (SSI) models, where the target depends on a single direction in input space applied to a sequence of tokens. This setting generalizes classical single-index models to the sequential domain, encompassing simplified one-layer attention architectures. We derive a closed-form expression for the population loss in terms of a pair of sufficient statistics capturing semantic and positional alignment, and characterize the induced high-dimensional SGD dynamics for these coordinates. Our analysis reveals two distinct training phases: escape from uninformative initialization and alignment with the target subspace, and demonstrates how the sequence length and positional encoding influence convergence speed and learning trajectories. These results provide a rigorous and interpretable foundation for understanding how sequential structure in data can be beneficial for learning with attention-based models.

CONTRIBUTION. I have designed the theoretical framework, not including the formal proofs. I have conducted all numerical experiments and illustrations.

A paper partially related to the thesis work, but not included in the thesis, is:

6. The Benefits of Reusing Batches for Gradient Descent in Two-Layer Networks: Breaking the Curse of Information and Leap Exponents

Yatin Dandi, Emanuele Troiani, Luca Arnaboldi, Luca Pesce, Lenka Zdeborová, Florent Krzakala

International Conference on Machine Learning (ICML) 2024

ABSTRACT. We investigate the training dynamics of two-layer neural networks when learning multi-index target functions. We focus on multi-pass gradient descent (GD) that reuses the batches multiple times and show that it significantly changes the conclusion about which functions are learnable compared to single-pass gradient descent. In particular, multi-pass GD with finite stepsize is found to overcome the limitations of gradient flow and single-pass GD given by the information exponent [BGJ21] and leap exponent [AAM23] of the target function. We show that upon re-using batches, the network achieves in just *two* time steps an overlap with the target subspace even for functions not satisfying the staircase property [Abb+21]. We characterize the (broad) class of functions efficiently learned in finite time. The proof of our results is based on the analysis of the Dynamical Mean-Field Theory (DMFT). We further provide a closed-form description of the dynamical process of the low-dimensional projections of the weights, and numerical experiments illustrating the theory.

CONTRIBUTION. I have contributed to the problem formulation and design. I have participated in the numerical implementation and executed a portion of the numerical experiments.

My collaborators and I have also explored a new promising line of research on the spectral theory of hierarchical feature learning in deep networks.

7. Deep Learning as Neural Low-Degree Filtering: A Spectral Theory of Hierarchical Feature Learning

Yatin Dandi, Matteo Vilucchio, Luca Arnaboldi, Hugo Tabanelli, Florent Krzakala

arXiv preprint 2026, under review at NeurIPS 2026

Abstract. Understanding how deep neural networks learn useful internal representations from data remains a central open problem in the theory of deep learning. We introduce *Neural Low-Degree Filtering* (Neural LoFi), a stylized limit of gradient-based training in which hierarchical feature learning becomes an explicit iterative spectral procedure. In this limit, the dynamics at each layer decouple: given the current representation, the next layer selects directions with maximal accessible low-degree correlation to the label. This yields a tractable surrogate mechanism for deep learning, together with a natural kernel-space interpretation. Neural LoFi provides a mathematically explicit framework for studying multi-layer feature learning beyond the lazy regime. It predicts how representations are selected layer by layer, explains how emergence of concepts arises with given sample complexity, and gives a concrete mechanism by which depth progressively constructs new features from old ones through low-degree compositionality. We complement the theory with mechanistic experiments on fully connected and convolutional architectures, showing that Neural LoFi improves over lazy random-feature baselines, recovers meaningful structured filters, and predicts representations aligned with early gradient-descent feature discovery with real datasets.

Contribution. I have contributed to the codebase for the numerical experiments, and executed a portion of them.

In the last two years of the doctoral program, we have started a collaboration on large language models for information retrieval, that ended up in a state of the art model on BEIR benchmark, in the class of models <150M parameters. This project was possible thanks to a grant awarded by the Swiss AI Initiative, for 26,000 GPU hours on the Swiss National Supercomputing Centre (CSCS) Clariden cluster.

The model was SOTA when released in January 2026, but has been surpassed since then.

8. ColBERT-Zero: To Pre-train Or Not To Pre-train ColBERT models

Antoine Chaffin, Luca Arnaboldi, Amélie Chatelain, Florent Krzakala
Late Interaction Workshop at European Conference on Information Retrieval (ECIR) 2026

Abstract.

Current state-of-the-art multi-vector models are obtained through a small Knowledge Distillation (KD) training step on top of strong single-vector models, leveraging the large-scale pre-training of these models. In this paper, we study the pre-training of multi-vector models and show that large-scale multi-vector pre-training yields much stronger multi-vector models. Notably, a fully ColBERT-pre-trained model, ColBERT-Zero, trained only on public data, outperforms GTE-ModernColBERT as well as its base model, GTE-ModernBERT, which leverages closed and much stronger data, setting new state-of-the-art for model this size. We also find that, although performing only a small KD step is not enough to achieve results close to full pre-training, adding a supervised step beforehand allows to achieve much closer performance while skipping the most costly unsupervised phase. Finally, we find that aligning the fine-tuning and pre-training setups is crucial when repurposing existing models. To enable exploration of our results, we release various checkpoints[1] as well as code used to train them[2].

Contribution. I was responsible for writing and deploying the training code on the cluster, running all ablations; I have also made some minor contributions to the underlying packages (`pylate`, `sentence-transformers`) that power the training code.

Finally, I was part of the creation of one of the most used benchmarks for measuring the reasoning capabilities of large language models, called *Humanity's Last Exam*.

[1] `https://huggingface.co/collections/lightonai/colbert-zero`
[2] `https://github.com/lightonai/pylate/tree/main/examples/train/ColBERT-zero/`

9. A benchmark of expert-level academic questions to assess AI capabilities

Center for AI Safety, Scale AI, HLE Contributors Consortium

Nature vol. 649, 2026

Abstract. Benchmarks are important tools for tracking the rapid advancements in large language model (LLM) capabilities. However, benchmarks are not keeping pace in difficulty: LLMs now achieve over 90% accuracy on popular benchmarks like MMLU, limiting informed measurement of state-of-the-art LLM capabilities. In response, we introduce Humanity's Last Exam (HLE), a multi-modal benchmark at the frontier of human knowledge, designed to be the final closed-ended academic benchmark of its kind with broad subject coverage. HLE consists of 2,500 questions across dozens of subjects, including mathematics, humanities, and the natural sciences. HLE is developed globally by subject-matter experts and consists of multiple-choice and short-answer questions suitable for automated grading. Each question has a known solution that is unambiguous and easily verifiable, but cannot be quickly answered via internet retrieval. State-of-the-art LLMs demonstrate low accuracy and calibration on HLE, highlighting a significant gap between current LLM capabilities and the expert human frontier on closed-ended academic questions. To inform research and policymaking upon a clear understanding of model capabilities, we publicly release HLE at this https URL[3].

Contribution. I have designed 2 questions that were included in the benchmark, that at the time of the release were not solvable by any LLM.

[3] `https://lastexam.ai/`

Contents

Abstract v

Sommario vii

Résumé ix

Acknowledgements xi

Foreword xiii

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Settings and terminology . . . 3
- 1.3 The Saad & Solla equations . . . 7
  - 1.3.1 The order-parameter dynamics . . . 7
  - 1.3.2 From discrete updates to ODEs . . . 8
  - 1.3.3 Anatomy of the equations: learning vs. noise . . . 10
  - 1.3.4 Closed-form examples . . . 11
  - 1.3.5 Beyond the high-dimensional limit . . . 12
- 1.4 Information exponent and beyond . . . 14
  - 1.4.1 Leap exponent . . . 19
  - 1.4.2 Generative exponent . . . 23

2 From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layers networks 27
- 2.1 Introduction . . . 27
- 2.2 Setting . . . 30
- 2.3 The three limit regimes and their dimensionless description . . . 31
  - 2.3.1 Main concepts . . . 32
  - 2.3.2 The classical regime . . . 33
  - 2.3.3 The high-dimensional regime . . . 34
  - 2.3.4 The overparametrized regime . . . 35
- 2.4 Conclusion . . . 40

3 Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD 41
3.1 Introduction . . . 41
3.2 High-dimensional limit of SGD . . . 44
3.3 Escaping mediocrity in the well-specified scenario . . . 47
3.3.1 Population risk landscape . . . 47
3.3.2 Exit time from deterministic limit . . . 48
3.3.3 Does stochasticity matter? . . . 49
3.4 The role of width . . . 51
3.4.1 Training the second layer . . . 53
3.5 Conclusion . . . 54

4 Online Learning and Information Exponents: On The Importance of Batch size, and Time / Complexity Tradeoffs 55
4.1 Introduction . . . 55
4.2 Setting, contributions, and related works . . . 56
4.3 Time / Complexity tradeoffs . . . 61
4.3.1 Weak recovery with one-pass SGD . . . 62
4.3.2 Illustration of Theorem 4.1 . . . 63
4.3.3 The self-interaction regime . . . 64
4.4 Exact asymptotic description . . . 67
4.4.1 Closed form equations . . . 67
4.5 Conclusions . . . 70

5 Repetita Iuvant: Data Repetition Allows SGD to Learn High-Dimensional Multi-Index Functions 73
5.1 Introduction . . . 73
5.2 Setting and Main Contributions . . . 75
5.3 Single-Index Model . . . 79
5.3.1 Main results . . . 79
5.3.2 Discussion . . . 81
5.3.3 Numerical illustrations . . . 81
5.4 Multi-Index Model . . . 83
5.4.1 Numerical investigation with multi-index target . . . 84
5.4.2 Learning hard functions through a hierarchical mechanism . . . 84
5.5 Conclusions . . . 88

6 Asymptotics of SGD in Sequence-Single Index Models and Single-Layer Attention Networks 89
6.1 Setting and definitions . . . 92
6.2 The sample complexity of SGD . . . 94
6.3 The role of the sequence length . . . 98
6.4 Positional encoding and training dynamics . . . 100
6.5 Conclusion . . . 104

Appendices **105**

A Appendices to the Introduction **105**
A.1 The single-index overlap dynamics on the sphere . . . 105
A.1.1 Derivation as a spherical gradient flow . . . 105
A.1.2 Equivalence with the Saad & Solla equation . . . 106
A.2 Validity window of the deterministic description . . . 107
A.3 The partial-trace algorithm . . . 108
A.3.1 The population identity . . . 108
A.3.2 The estimator . . . 110
A.3.3 Why $n \asymp d^{\ell/2}$ is the threshold . . . 110
A.3.4 Familiar special cases . . . 111

B Appendices to "From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layers networks" **113**
B.1 Effect of noise on Equations (2.14) . . . 113
B.2 Derivation of Equation (2.21) and local fields covariance . . . 114
B.3 Mean-field approximation: proof of Theorem 2.2 . . . 115
B.4 Proofs for the high-dimensional mean-field approximation . . . 119
B.4.1 Preliminaries . . . 119
B.4.2 Proof of Lemma 2.1 . . . 120
B.4.3 Proof of Equation (2.29) . . . 121
B.5 Derivation of explicit expression for the squared activation . . . 122
B.6 Numerical experiments in the mean-field limit . . . 124

C Appendices to "Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD" **127**
C.1 Explicit ODEs derivation . . . 127
C.1.1 Derivation of the process . . . 127
C.1.2 Explicit ODE . . . 128
C.2 Spherically constrainted ODE and SDE . . . 129
C.2.1 Spherical constraint for ODE . . . 129
C.2.2 Spherical constraint for SDE . . . 132
C.3 Derivation of the expected exit time formulas . . . 133
C.3.1 Linearization of the equations . . . 133
C.3.2 Solving the approximated risk equation . . . 134
C.3.3 Averaging on initial conditions . . . 135
C.3.4 Overparameterization Gain . . . 136
C.4 Stochastic correction to the exit-time formula . . . 136
C.5 SDEs for arbitrary width . . . 137
C.5.1 Unconstrained SDE with $p > 1$ . . . 138
C.5.2 Spherical Constrained SDE . . . 138
C.5.3 Special cases . . . 140
C.6 Stochastic correction from [BGJ22] . . . 141

C.6.1 Diffusion at generic point . . . 144
C.7 Landscape geometry . . . 146
C.7.1 Landscape in the sphere . . . 149

D Appendices to "Online Learning and Information Exponents: On The Importance of Batch size, and Time / Complexity Tradeoffs" 151
D.1 Proof of Theorems 4.1 and 4.2 . . . 151
D.1.1 Preliminaries . . . 151
D.1.2 Computing the gradient at time t . . . 153
D.1.3 A differential inequality for m_t . . . 154
D.1.4 From the linear regime to one-step recovery . . . 159
D.2 Informal derivation of Proposition 4.1 . . . 161
D.3 Special case: committee machine with matching architecture . . . 165
D.3.1 Analytic case $\sigma = \mathrm{erf}\left(\cdot/\sqrt{2}\right)$ . . . 167
D.3.2 Analytic case $\sigma = \mathrm{He}_2$ . . . 168
D.3.3 Analytic case $\sigma = \mathrm{He}_3$ . . . 168
D.4 Weak recovery with Generalized Linear Models . . . 170
D.4.1 Correlations loss SGD . . . 172
D.4.2 Simple example: retrieving [BGJ21] . . . 172
D.4.3 Extension to $p > 1$ . . . 172
D.5 Additional numerical investigation . . . 173
D.5.1 Cold start for multi-index models . . . 173
D.5.2 Behavior of *Spherical SGD* . . . 174
D.5.3 *Adaptive SGD*: combining Correlation Loss SGD with plain SGD . . . 175
D.5.4 Polylog regime example . . . 176
D.5.5 Large-batch corrections to asymptotic dynamics . . . 177

E Appendices to "Repetita Iuvant: Data Repetition Allows SGD to Learn High-Dimensional Multi-Index Functions" 179
E.1 Theoretical framework . . . 179
E.1.1 Difference inequalities for vector-valued processes . . . 179
E.1.2 Low-dimensional sufficient statistics . . . 182
E.1.3 Bounding the hitting times . . . 185
E.2 Proof of the Main Results . . . 188
E.2.1 Proof of Theorem 5.2 . . . 188
E.2.2 Proof of Theorem 5.4 . . . 191
E.3 Additional proofs . . . 195
E.3.1 Proof of Lemma 5.1 . . . 195
E.3.2 Proof of Theorem 5.3 . . . 195
E.3.3 Hardness of sign functions . . . 196
E.4 Additional Numerical Investigation . . . 197
E.4.1 Testing wider range of Algorithm 5.1 . . . 198
E.4.2 Extensive batch size . . . 198
E.5 Implementation Details . . . 201

E.5.1 Hyperparameters of the Figures 201

F Appendices to "Asymptotics of SGD in Sequence-Single Index Models and Single-Layer Attention Networks" 203
F.1 Reduction from attention to sequence single-index 203
F.2 Mathematical preliminaries and notations 204
F.2.1 Tensors 204
F.2.2 Hermite Polynomials 205
F.3 Formalization and proofs 207
F.3.1 Preliminaries 207
F.3.2 An ODE for overlap evolution 207
F.3.3 Controlling the gradient norm 209
F.3.4 Hitting time for the tied dynamics 210
F.3.5 Hitting time for untied dynamics 211
F.3.6 The effect of positional encoding on the flatness 212
F.4 Sequence Information Exponent Beyond attention 213
F.5 Further analysis on the effect of sequence length 213
F.5.1 Anatomy of the gain 214
F.5.2 Example at non-optimal learning rate 216
F.5.3 The upper bound on the learning rate scaling 216
F.5.4 Equivalence between Single-Layer attention and Tied network 218
F.5.5 Pathological cases 219
F.6 Further discussion on the positional-semantic transition 220
F.6.1 SIE of model (6.18) 220
F.6.2 Numerical experiments on the transition 222
F.6.3 An alternative model without phase transition 223
F.7 Numerical experiments details 224
F.7.1 Details on the integration method of squared loss 224

Bibliography 227

Curriculum Vitae 241

Chapter 1

Introduction

1.1 Motivation

Artificial Intelligence (AI) is the buzzword of this decade. The release of ChatGPT in late 2022 triggered a wave of public interest in machine learning that had been building for years in research laboratories. What made this transition possible was the solution of an extraordinary engineering challenge: the construction of a pipeline capable of digesting nearly the entirety of human digital knowledge and using it to train deep learning models of unprecedented scale. This achievement draws on decades of cumulative work spanning hardware design, software engineering, and fundamental research: from the first attempts to emulate the human brain with artificial neural networks in the 1950s, through the development of backpropagation in the 1980s, to the rise of GPU computing in the 2000s and the explosion of large-scale datasets in the 2010s. Yet for all this engineering progress, our theoretical understanding of *why* these models work remains strikingly limited.

The reasons to close this gap are manifold. The most immediate are practical: a deeper understanding can guide the design of better algorithms, improve the efficiency of existing models, and reduce the enormous computational and energy costs of training. The social motivations are equally compelling: a rigorous theory underpins the growing fields of interpretability, fairness, and robustness, which are essential if AI systems are to be deployed safely in high-stakes settings. But this thesis is submitted to the *École Doctorale de Physique* (EDPY), and physicists are not driven by practical concerns alone. We are also moved by the desire to understand the fundamental principles governing the behavior of complex systems. There is something deeply human in wanting to understand the mechanisms of a technology that increasingly surrounds us and shapes every aspect of our lives; satisfying that curiosity is a legitimate scientific goal in its own right, and has historically led to unexpected discoveries with far-reaching consequences.

What makes deep learning theoretically challenging is precisely what makes it empirically powerful. Many machine learning models are theoretically grounded by construction: support vector machines rest on the geometry of convex optimization; decision trees can be analyzed with combinatorial and probabilistic tools. Deep learning broke free of these clean frameworks. The central

idea, using a very large number of parameters to fit data, violates the assumptions underlying classical statistics: the bias–variance trade-off, convexity, and the availability of closed-form solutions. Yet deep networks learn effective representations, generalize remarkably well to unseen examples, and are trained with algorithms as simple as stochastic gradient descent navigating a highly non-convex loss landscape. This has challenged many of the intuitions we had about learning and generalization, and continues to generate vigorous debate in the research community.

A physicist's instinct, confronted with a phenomenon that outstrips existing theory, is to build a model, a simplified but principled caricature that retains the essential features and can be understood analytically. Think of where chemistry stood at the end of the nineteenth century: practitioners could synthesize remarkably complex molecules and predict reactions with considerable reliability, yet nobody understood the deep mechanism, why atoms bond the way they do. The Thomson model was the first crude, almost certainly incomplete picture of the underlying atomic structure. It was wrong in important ways, but concrete enough to be tested, refuted, and improved. This is, roughly speaking, where we stand with deep learning today, and it is the role we aspire to play. We want to be Thomson, so that someone can be Bohr after us, and eventually someone arrives at quantum mechanics.

One advantage we have is that our experiments run on computers. We do not need a particle collider to test a theory of deep learning: we need a laptop (or a cluster for large-scale experiments) and a few days of compute time. This is an enormous freedom, and we should exploit it. A second advantage is that the fundamental process at the core is mathematically well-defined: training a neural network is an optimization procedure whose rules are written down precisely, even if their consequences are not. We do not need to figure out *what* the process is; only *why* it works and *how* its dynamics unfold. The goal is a theory that is both analytically tractable and grounded in controlled experiments. The settings we analyze remain idealizations, and closing the gap to real-world systems is the longer goal; here we take one small, deliberate step toward it.

Among the many aspects of deep learning that resist theoretical explanation, we focus on the *dynamics of training*: the process by which randomly initialized parameters converge to a configuration that generalizes well to unseen data. The loss landscape of deep learning is highly non-convex, and it is far from obvious why stochastic gradient descent should find good solutions in it. Yet it does, and the dynamics induced by SGD introduce an inductive bias that correlates with generalization. Understanding this bias, its origins, and its consequences is the central thread running through this thesis.

Classical learning theory—PAC learning, VC dimension, Rademacher complexity and all this jargon of tools—analyzes generalization in the worst case over all possible data distributions. This regime is useful as a foundation but tends to produce pessimistic bounds that badly overestimate the true error of well-trained deep networks. Statistical physics offers a complementary perspective, analyzing the *typical* behavior of learning algorithms averaged over a natural distribution of instances. This leads to sharper, more realistic predictions. The connection between statistical physics and machine learning is not new: the tools of disordered systems and mean-field theory were applied to neural network capacity and generalization as far back as the 1980s. In some sense, machine learning is a canonical problem of statistical physics: large numbers of parameters, large datasets, and emergent collective behavior. We inherit and build on this tradition.

1.2 Settings and terminology

With these motivations in place, we now describe the concrete setting that serves as our laboratory throughout the thesis. We work in *supervised learning with scalar labels*: given a dataset of input–output pairs, the goal is to learn a function that predicts the label from the input. Among all learning paradigms, supervised regression is the one that affords the most control: we need not rely on the data's complex internal structure to build our model, and a scalar label carries more signal than a class label. This level of control is precisely what a physicist needs to build and test a theory.

For any supervised learning algorithm we need three ingredients: a *data distribution*, a *model class*, and a *learning algorithm*. Each ingredient offers enormous freedom, and our choices are guided by a single criterion: retain only the features that are essential for the phenomena we wish to study, and strip everything else.

What data do we need? Real-world data—images, text, audio—has rich statistical structure that is not yet amenable to exact analysis. We therefore replace it with the simplest distribution that still supports non-trivial learning: *isotropic Gaussian* inputs. Concretely, $\boldsymbol{x} \in \mathbb{R}^d$ is drawn i.i.d. from

$$\boldsymbol{x} \sim \mathcal{N}(0, I_d).$$

Obviously, this is a strong assumption that does not reflect the complexity of real data. The main analytical payoff is that any linear transformation of a Gaussian vector is again Gaussian. As we will see, this turns a d-dimensional stochastic problem into a low-dimensional one. Although real data is rarely Gaussian, Gaussian universality justifies it as a tractable proxy for studying SGD dynamics. Goldt et al. [Gol+20] show, via the *Gaussian Equivalence Property* – later established rigorously by Goldt et al. [Gol+22] – that for data drawn from a one-layer generative model (the hidden-manifold model) the pre-activations driving online SGD become jointly Gaussian in the high-dimensional limit, with the data entering only through its first two moments; the resulting learning dynamics then coincide with those of an equivalent Gaussian model. We therefore adopt i.i.d. Gaussian inputs, not as a literal data model but as the canonical representative of a universality class capturing the relevant high-dimensional dynamics. There have been some recent efforts to relax this assumption: Zweig, Pillaud-Vivien, and Bruna [ZPB23] generalize some single-index results to spherically symmetric distributions, while Cagnetta et al. [Cag+24] introduce a different hierarchical model of the data. In *Independent Component Analysis* (ICA) the departure from Gaussianity is not a nuisance but the very quantity to be recovered: the hidden directions surface only through the higher-order cumulants that a Gaussian lacks [RBG25]. The Gaussian case nonetheless remains the most tractable and best understood, thus it is the one we adopt in this thesis.

We still need a label y for each input $\boldsymbol{x}$. We take it to be generated by a fixed *teacher* function $f^\star$, with optional additive Gaussian noise:

$$y = f^\star(\boldsymbol{x}) + \sqrt{\Delta}\,\xi, \qquad \xi \sim \mathcal{N}(0,1), \tag{1.1}$$

where $\Delta \geq 0$ is the noise variance and $\Delta = 0$ is the *realizable* case; we add this to simulate the noise and imperfections of real data. The teacher models the hidden rule that generates the data.

We would like it to be non-trivial but not arbitrary; the right compromise is to suppose that the labels depend on the input only through a few directions—the canonical form of low-dimensional structure. More precisely, we take the teacher to be a *multi-index model*:

$$f^\star(\boldsymbol{x}) = h^\star\left(\frac{W^\star \boldsymbol{x}}{\sqrt{d}}\right), \tag{1.2}$$

where $W^\star = \{\boldsymbol{w}_r^\star\}_{r\in[k]} \in \mathbb{R}^{k\times d}$ is the *teacher weight matrix* whose $k < d$ rows $\boldsymbol{w}_r^\star$ are the *teacher directions*, and $h^\star : \mathbb{R}^k \to \mathbb{R}$ is a possibly nonlinear *link function*. The special case $k = 1$ is the *single-index model*, with a single relevant direction $\boldsymbol{w}^\star$. The subspace $V^\star = \mathrm{span}(\boldsymbol{w}_1^\star, \dots, \boldsymbol{w}_k^\star)$ in a d-dimensional space is the *teacher subspace*. These $k \ll d$ directions *are* the features of the problem: they are the only aspect of the input on which the label depends, and the entire learning process can be thought of as the discovery of this subspace and the alignment of the student's weights with it. This is precisely what *feature learning* means here, in an unambiguous and measurable sense, and it is what makes the multi-index model the right laboratory for the rest of the thesis: the difficulty of recovering $V^\star$ is exactly what the information, leap, and generative exponents of Section 1.4 will quantify.

The normalization of the teacher directions plays an important role in the definition of all the subsequent quantities, and different conventions are used in the literature. We take them to be standard Gaussian, $\boldsymbol{w}_r^\star \sim \mathcal{N}(0, I_d)$, so that $\|\boldsymbol{w}_r^\star\| \approx \sqrt{d}$, and state this convention explicitly whenever it is relevant.

What model do we use? Deep learning is a broad family of models, often designed for a specific data modality or task. Yet, the core of deep learning is the same across all these variations: stacked layers of linear transformations and nonlinear activations. We take its simplest non-trivial representative: a *two-layer (shallow) neural network* with p hidden neurons, first-layer weights $W = \{\boldsymbol{w}_j\}_{j\in[p]} \in \mathbb{R}^{p\times d}$, second-layer weights $\boldsymbol{a} = (a_1, \dots, a_p)^\top \in \mathbb{R}^p$, and pointwise activation function $\sigma : \mathbb{R} \to \mathbb{R}$:

$$f(\boldsymbol{x};\, W, \boldsymbol{a}) = \frac{1}{p}\sum_{j=1}^{p} a_j\, \sigma\left(\frac{\langle \boldsymbol{w}_j, \boldsymbol{x}\rangle}{\sqrt{d}}\right). \tag{1.3}$$

Figure 1.1 contrasts this student network with the multi-index teacher it is trained to imitate.

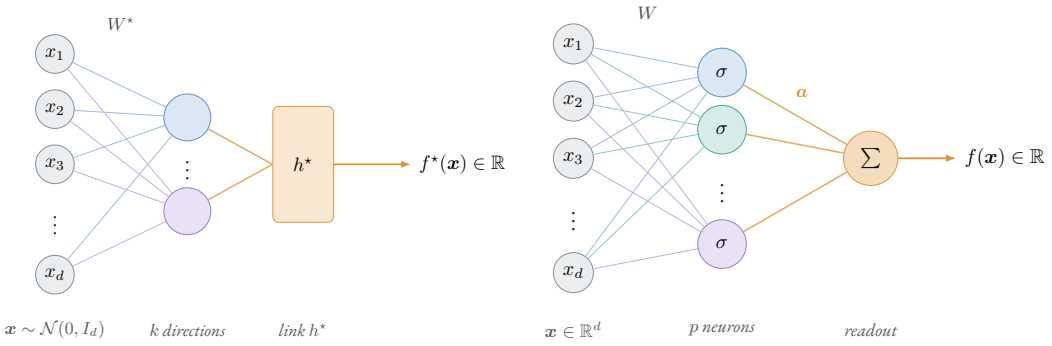
Two-layer multi-index networks are just one among infinitely many possible ways to proxy deep learning. As non-exhaustive examples, in the literature we find deep linear networks [SMG14; ACH18], infinite-depth-and-width limits with DMFT [Bor+24], and tensor programs [Yan19]. The approach we discuss is not the only one for two-layer networks either: diagonal linear networks [PPF21; Eve+23], the mean-field limit of two-layer nets [MMN18; CB18; RV22; SS20], the neural tangent kernel / lazy regime [JGH18], and the landscape geometry of linear networks [TKB20].

The $1/p$ prefactor is the *mean-field normalization*, which keeps the output finite as $p \to \infty$. The student parameters are $\Theta = (\boldsymbol{a}, W)$. Crucially, Equation (1.3) has the same functional dependence on a low-dimensional projection of the input as the multi-index teacher (1.2): the student is a special case of the teacher. This symmetry is why the setting is called the *teacher–student scenario*. The two-layer network is the simplest architecture that generates a non-convex loss landscape; already with $k = p = 1$ and a nonlinear activation, the landscape has saddle points and spurious local minima that make the dynamics of gradient descent non-trivial.

Think of *phase retrieval* ($\sigma(z) = z^2$): the loss landscape has at least two global minima, at $\boldsymbol{w} = \pm\boldsymbol{w}^\star$, and a saddle point at $\boldsymbol{w} = 0$.

To measure the quality of the prediction of the model, we use the quadratic (*square*) loss

$$\ell(\hat{y}, y) = \tfrac{1}{2}(\hat{y} - y)^2$$



(a) multi-index teacher (b) two-layer student

Figure 1.1: The teacher–student scenario. **(a)** The multi-index teacher (1.2): the Gaussian input $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$ is projected onto the $k < d$ teacher directions $W^\star$ (each drawn in its own color) and passed jointly through a general nonlinear link $h^\star$ to produce the label. **(b)** The two-layer student (1.3): each of the p hidden neurons applies the *pointwise* activation σ to a linear readout $\langle \boldsymbol{w}_j, \boldsymbol{x} \rangle / \sqrt{d}$ through its first-layer weights $\boldsymbol{w}_j$ (a row of W, each neuron in its own color), and the second-layer weights $\boldsymbol{a}$ combine them with the mean-field normalization $1/p$. The student is the special case in which the teacher's link is a sum of pointwise activations.

throughout, and measure the overall performance with *generalization error* or *population risk*,

$$\mathcal{R}(W, \boldsymbol{a}) = \mathbb{E}_{(\boldsymbol{x}, y)} \left[\frac{1}{2} \big(f(\boldsymbol{x};\, W, \boldsymbol{a}) - y \big)^2 \right]. \tag{1.4}$$

Note that the expectation is taken over the data distribution, not the training set; this is what makes it a measure of generalization, since we know the precise data distribution. The training happens on samples drawn from the data distribution, and not on the population risk, which is not accessible to the learning algorithm.

The square loss—predicting a scalar $y \in \mathbb{R}$—is especially convenient: its gradient is proportional to the residual error, the population risk admits a closed-form expression in a small set of order parameters (defined below), and the resulting loss landscape, while non-convex, is structured enough to be studied rigorously.

What algorithm do we use? Stochastic gradient descent (SGD) is the workhorse optimizer of modern deep learning: virtually every model in practice is trained with it or one of its descendants. Our proxy for SGD is *online (one-pass) SGD*: at each step ν a fresh, independently drawn sample $(\boldsymbol{x}^\nu, y^\nu)$ is used to form a gradient estimate and update the first-layer weights,

$$\boldsymbol{w}_j^{\nu+1} = \boldsymbol{w}_j^\nu - \gamma \, \nabla_{\boldsymbol{w}_j} \, \ell\big(f(\boldsymbol{x}^\nu;\, W^\nu, \boldsymbol{a}),\, y^\nu \big), \qquad j \in [p], \tag{1.5}$$

with learning rate $\gamma > 0$. The number of steps n equals the number of samples consumed. Modern practice uses mini-batch multi-epoch training with sophisticated optimizers (Adam, Muon, etc.) on datasets that are revisited many times. Ours is simpler in the one key respect that makes

it analytically tractable: the *independence* of successive gradient steps. This independence is also not as artificial as it might seem: when the dataset is large enough that each sample is seen at most once, the training regime is effectively online.

We relax the independence assumption in Chapter 5, and lifting it turns out to be a pivotal step in the theory.

Finally, we initialize the weights randomly: $\boldsymbol{w}_j^0 \sim \mathcal{N}(0, I_d)$.

Order parameters. We now come to the central observation that makes the whole program analytically tractable: both teacher and student see the input only through a handful of one-dimensional projections, the *pre-activations* (or *local fields*)

$$\boldsymbol{\lambda} = \frac{W\boldsymbol{x}}{\sqrt{d}} \in \mathbb{R}^p, \qquad \boldsymbol{\lambda}^\star = \frac{W^\star \boldsymbol{x}}{\sqrt{d}} \in \mathbb{R}^k, \tag{1.6}$$

where W and $W^\star$ are the student and teacher weight matrices respectively (defined above). Indeed, the student (1.3) depends on the input only through its own pre-activations: writing $\lambda_j = \langle \boldsymbol{w}_j, \boldsymbol{x}\rangle/\sqrt{d}$, we have

$$f(\boldsymbol{x};\, W, \boldsymbol{a}) = h(\boldsymbol{\lambda}, \boldsymbol{a}), \qquad h(\boldsymbol{\lambda}, \boldsymbol{a}) = \frac{1}{p}\sum_{j=1}^{p} a_j\, \sigma(\lambda_j), \tag{1.7}$$

which mirrors exactly the teacher $f^\star(\boldsymbol{x}) = h^\star(\boldsymbol{\lambda}^\star)$ from (1.2).

Because the input is Gaussian and the pre-activations are linear in $\boldsymbol{x}$, the pair $(\boldsymbol{\lambda}, \boldsymbol{\lambda}^\star)$ is *jointly Gaussian*, with zero mean and a covariance fixed entirely by the weight matrices, not by any individual data point. That covariance is assembled from the pairwise inner products of the student and teacher rows, which we collect into the *order parameters* (or *overlap matrices*):

$$Q_{jl} = \frac{\langle \boldsymbol{w}_j,\, \boldsymbol{w}_l\rangle}{d}, \qquad M_{jr} = \frac{\langle \boldsymbol{w}_j,\, \boldsymbol{w}_r^\star\rangle}{d}, \qquad P_{rs} = \frac{\langle \boldsymbol{w}_r^\star,\, \boldsymbol{w}_s^\star\rangle}{d}. \tag{1.8}$$

The matrix $Q \in \mathbb{R}^{p\times p}$ encodes the self-correlations among student neurons; $M \in \mathbb{R}^{p\times k}$ measures the alignment of each student neuron with each teacher direction; and $P \in \mathbb{R}^{k\times k}$ describes the geometry of the teacher (concentrating on $P = I_k$ in the high-dimensional limit under our Gaussian normalization of the teacher directions). Together they form the covariance of the pre-activations,

$$(\boldsymbol{\lambda}, \boldsymbol{\lambda}^\star) \sim \mathcal{N}(0, \Omega), \qquad \Omega = \begin{pmatrix} Q & M \\ M^\top & P \end{pmatrix} \in \mathbb{R}^{(p+k)\times(p+k)}. \tag{1.9}$$

The triple $(Q, M, \boldsymbol{a})$—equivalently $(\Omega, \boldsymbol{a})$, since P is fixed—is therefore a set of *sufficient statistics* for the problem: every quantity that depends on the weights only through the joint law of the pre-activations is a function of it alone.

The population risk is exactly such a quantity. Substituting the student (1.7) into the risk (1.4) and using the joint Gaussianity (1.9), the expectation over $\boldsymbol{x} \in \mathbb{R}^d$ collapses to an expectation over the $(p+k)$-dimensional Gaussian $\mathcal{N}(0, \Omega)$:

$$\mathcal{R}(\Omega, \boldsymbol{a}) = \mathbb{E}_{(\boldsymbol{\lambda},\, \boldsymbol{\lambda}^\star)\sim\mathcal{N}(0,\Omega)}\left[\frac{1}{2}\big(h(\boldsymbol{\lambda}, \boldsymbol{a}) - h^\star(\boldsymbol{\lambda}^\star)\big)^2\right]. \tag{1.10}$$

The crucial feature of this expression is that it *does not scale with* d: it involves only the $O(p^2+pk)$ entries of $(\Omega, \boldsymbol{a})$, with the ambient dimension entering nowhere. Rather than tracking the $p \times d$ weight vectors and an expectation over $\mathbb{R}^d$, we track a handful of scalar overlaps and integrate a fixed low-dimensional Gaussian. It is precisely this dimension-independence that lets us send $d \to \infty$, the core missing piece of our analysis.

1.3 The Saad & Solla equations

We now turn to the result from the literature on which much of this thesis is built. Compressing the problem to the order parameters $(Q, M, \boldsymbol{a})$ is only half of the story; the other half is the *high-dimensional limit* $d \to \infty$. In this regime the stochastic fluctuations of the order parameters around their mean become negligible, and the entire training trajectory concentrates on a deterministic path governed by a system of *ordinary differential equations* (ODEs).

1.3.1 The order-parameter dynamics

We can make Equation (1.5) explicit in terms of the local fields, and write the stochastic dynamics of the weights $\boldsymbol{w}_j$ and the output weights a_j of the student network as

$$\boldsymbol{w}_j^{\nu+1} = \boldsymbol{w}_j^{\nu} + \frac{\gamma}{p\sqrt{d}} \left(y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^{\nu})\right) a_j \sigma'(\lambda_j^{\nu}) \boldsymbol{x}^{\nu}, \tag{1.11}$$

$$a_j^{\nu+1} = a_j^{\nu} + \frac{\gamma}{pd} \left(y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^{\nu})\right) \sigma(\lambda_j^{\nu}) \tag{1.12}$$

where $\boldsymbol{\lambda}^{\nu} = \frac{1}{\sqrt{d}}(\langle \boldsymbol{w}_1, \boldsymbol{x}^{\nu}\rangle, \dots, \langle \boldsymbol{w}_p, \boldsymbol{x}^{\nu}\rangle)$ and $\boldsymbol{\lambda}^{\star,\nu} = \frac{1}{\sqrt{d}}(\langle \boldsymbol{w}_1^{\star}, \boldsymbol{x}^{\nu}\rangle, \dots, \langle \boldsymbol{w}_k^{\star}, \boldsymbol{x}^{\nu}\rangle)$ are the *local fields* (i.e., pre-activations) of the student and teacher networks, respectively, relative to the current input $\boldsymbol{x}^{\nu}$ used in the ν-th SGD step. The quantity $y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) \coloneqq y(\boldsymbol{\lambda}^{\star,\nu}) + \sqrt{\Delta}\xi^{\nu}$ is the noisy label of the current training sample. The second-layer weights a_j are updated with a rescaled learning rate γ/d, which keeps both layers evolving on the same timescale as $d \to \infty$. With a little extra work we can derive the update rule for the order parameters themselves,

$$\begin{aligned}
Q_{jl}^{\nu+1} =& Q_{jl}^{\nu} + \frac{\gamma}{pd} \left(y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^{\nu})\right) \left(a_l \sigma'(\lambda_l^{\nu})\lambda_j^{\nu} + a_j \sigma'(\lambda_j^{\nu})\lambda_l^{\nu}\right) + \\
&+ \frac{\gamma}{pd}\frac{\gamma}{p} \left(y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^{\nu})\right)^2 a_j a_l \sigma'(\lambda_j^{\nu})\sigma'(\lambda_l^{\nu}) \frac{\|\boldsymbol{x}^{\nu}\|^2}{d}, \\
M_{jr}^{\nu+1} =& M_{jr}^{\nu} + \frac{\gamma}{pd} \left(y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^{\nu})\right) a_j \sigma'(\lambda_j^{\nu}) \lambda_r^{\star,\nu}, \\
a_j^{\nu+1} =& a_j^{\nu} + \frac{\gamma}{pd} \left(y^{\nu}(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^{\nu})\right) \sigma(\lambda_j^{\nu}).
\end{aligned} \tag{1.13}$$

Since we are interested in the high-dimensional limit, the factor $\|\boldsymbol{x}^{\nu}\|^2/d$ in the Q update can be replaced by its expectation $\mathbb{E}\left[\|\boldsymbol{x}^{\nu}\|^2/d\right] = 1$. The update rules above are still stochastic, but they no longer depend explicitly on the input $\boldsymbol{x}^{\nu}$: they involve only the local fields $\boldsymbol{\lambda}^{\nu}$ and $\boldsymbol{\lambda}^{\star,\nu}$, which are Gaussian random variables whose covariance is given by the order parameters themselves. In other words, we have a *duality* between the process in the space of the weights $W \in \mathbb{R}^{p\times d}$ and

This is the classical *à la physique* argument from Saad and Solla [SS95a]. In the next section we will report a formalization of this argument.

the process in the space of the order parameters $(Q, M, \boldsymbol{a}) \in \mathbb{R}^{p\times p} \times \mathbb{R}^{p\times k} \times \mathbb{R}^p$

$$\begin{aligned} \text{update rule for } W \quad &\leftrightarrow \quad \text{update rule for } (Q, M, \boldsymbol{a}) \\ \text{sample } \boldsymbol{x}^\nu \sim \mathcal{N}(0, I_d) \quad &\leftrightarrow \quad \text{sample } (\boldsymbol{\lambda}^\nu, \boldsymbol{\lambda}^{\star,\nu}) \sim \mathcal{N}(0, \Omega) \\ \text{population risk } \mathcal{R}(W) \quad &\leftrightarrow \quad \text{population risk } \mathcal{R}(Q, M, \boldsymbol{a}). \end{aligned}$$

This duality gives a full characterization of the random SGD process in dimension d that is itself independent of d. The population risk is a quantity that fully characterizes the generalization properties of the student network, and it can be computed directly from the order parameters; the dynamics, in turn, is self-contained in the order-parameter space, so we can study the generalization of the student without ever dealing with the high-dimensional weights W directly.

1.3.2 From discrete updates to ODEs

The update rules in Equation (1.13) can be rearranged as

$$\begin{aligned} \frac{Q_{jl}^{\nu+1} - Q_{jl}^{\nu}}{\gamma/pd} &= \left(y^\nu(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^\nu)\right)\left(a_l\sigma'(\lambda_l^\nu)\lambda_j^\nu + a_j\sigma'(\lambda_j^\nu)\lambda_l^\nu\right) + \\ &\quad + \frac{\gamma}{p}\left(y^\nu(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^\nu)\right)^2 a_j a_l \sigma'(\lambda_j^\nu)\sigma'(\lambda_l^\nu), \\ \frac{M_{jr}^{\nu+1} - M_{jr}^{\nu}}{\gamma/pd} &= \left(y^\nu(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^\nu)\right) a_j \sigma'(\lambda_j^\nu), \\ \frac{a_j^{\nu+1} - a_j^{\nu}}{\gamma/pd} &= \left(y^\nu(\boldsymbol{\lambda}^{\star,\nu}) - h(\boldsymbol{\lambda}^\nu)\right)\sigma(\lambda_j^\nu). \end{aligned} \tag{1.14}$$

Intuitively, when $d \to \infty$ the left-hand side of the equations above can be interpreted as a derivative with respect to a continuous time variable $t = \nu\frac{\gamma}{pd}$, while the right-hand side can be replaced by its expectation over the Gaussian distribution of the local fields, which is parametrized by the order parameters themselves. This leads to a system of ODEs for the order parameters, known as

In the traditional derivation of the Saad & Solla equations, the usual time scaling is $t = \frac{\nu}{d}$. We will later clarify why we prefer this scaling.

the *Saad & Solla equations* [SS95a; SS96]:

$$
\begin{aligned}
\frac{\mathrm{d}Q_{jl}}{\mathrm{d}t} &= \overbrace{\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(0,\Omega)}\left[\left(h^\star(\boldsymbol{\lambda}^\star)-h(\boldsymbol{\lambda})\right)\left(a_l\sigma'(\lambda_l)\lambda_j + a_j\sigma'(\lambda_j)\lambda_l\right)\right]}^{=:\,\Phi^Q_{jl}(\Omega,a)} \\
&\quad + \frac{\gamma}{p}\underbrace{\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(0,\Omega)}\left[\left(\left(h^\star(\boldsymbol{\lambda}^\star)-h(\boldsymbol{\lambda})\right)^2+\Delta\right)a_j a_l\sigma'(\lambda_j)\sigma'(\lambda_l)\right]}_{=:\,\Psi^Q_{jl}(\Omega,a)}, \\
\frac{\mathrm{d}M_{jr}}{\mathrm{d}t} &= \underbrace{\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(0,\Omega)}\left[\left(h^\star(\boldsymbol{\lambda}^\star)-h(\boldsymbol{\lambda})\right)a_j\sigma'(\lambda_j)\lambda^\star_r\right]}_{=:\,\Phi^M_{jr}(\Omega,a)}, \\
\frac{\mathrm{d}a_j}{\mathrm{d}t} &= \underbrace{\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(0,\Omega)}\left[\left(h^\star(\boldsymbol{\lambda}^\star)-h(\boldsymbol{\lambda})\right)\sigma(\lambda_j)\right]}_{=:\,\Phi^a_j(\Omega,a)}.
\end{aligned}
\tag{SS-ODEs}
$$

Once the initial conditions are fixed, the trajectory drawn by these equations is completely determined: the dynamics is deterministic, and the randomness of the training process has been averaged out in the high-dimensional limit.

Remark 1.1. *Note that, even though the Saad & Solla equations determine a deterministic trajectory for the order parameters, the actual trajectory of the weights W is still stochastic, and impossible to characterize without knowing the specific realization of the training data $\boldsymbol{x}^\nu$. The Saad & Solla equations only describe the trajectory of the order parameters, and consequently of the population risk.*

The formalization of the argument above, together with the proof of convergence of the stochastic process to the deterministic one, was first given by Goldt et al. [Gol+19].

The original theorem was stated for matching student and teacher networks, but the proof extends straightforwardly to the more general case of a multi-index teacher and a general student network.

Theorem 1.1 (Convergence of the Saad & Solla equations, [Gol+19])**.** *Suppose that*

(i) The function σ is bounded, and its derivatives up to the second order exist and are bounded too;

(ii) The function $h^\star$ grows at most like a polynomial, i.e., $|h^\star(\boldsymbol{\lambda}^\star)| \leq K(1+\|\boldsymbol{\lambda}^\star\|^\alpha)$ for some constants K and α;

(iii) $W^\star$ is of rank k;

(iv) The initial macroscopic state Ω^0 is deterministic and bounded by a constant;

(v) The constants Δ, p, k and γ are all finite.

Then, for any $T>0$, the overlap matrix satisfies

$$
\max_{0\leq\nu\leq Td}\mathbb{E}\left[\left\|(\Omega^\nu,a^\nu)-\left(\tilde{\Omega}\left(\frac{\nu\gamma}{pd}\right),\tilde{a}\left(\frac{\nu\gamma}{pd}\right)\right)\right\|\right] \leq \frac{C(T)}{\sqrt{d}}, \tag{1.15}
$$

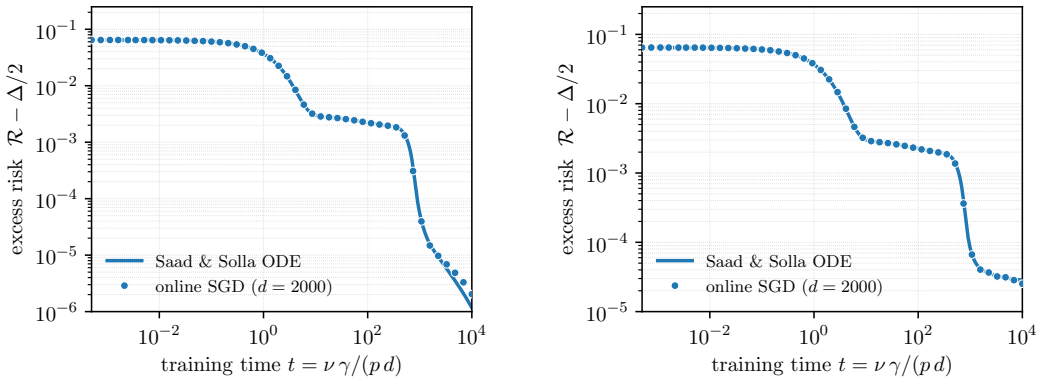


Figure 1.2: Excess risk $\mathcal{R} - \Delta/2 = \mathbb{E}[\frac{1}{2}(h - h^\star)^2]$ against training time $t = \nu\gamma/(pd)$ for the erf soft-committee machine ($p = 8$, $k = 4$, $\gamma = 2$, $d = 2000$). *Left* ($\Delta = 0$): the order parameters align with the teacher and the excess risk decreases monotonically toward zero. *Right* ($\Delta = 10^{-3}$): the variance term Ψ, proportional to γ, pins the dynamics at a finite noise plateau even as $t \to \infty$. The solid line is the deterministic Saad & Solla ODE; dots are a single online-SGD run at finite d, started from the same initial condition.

where $\left(\tilde{\Omega}, \tilde{a}\right)$ *is the unique solution of Equations* (SS-ODEs)*, with the initial condition*

$$\left(\tilde{\Omega}(0), \tilde{a(0)}\right) = \left(\Omega^0, a^0\right).$$

1.3.3 Anatomy of the equations: learning vs. noise

It is worth interpreting the actual content of the Saad & Solla equations; for this we follow the nomenclature used by Veiga et al. [Vei+22]. There are two types of terms:

(i) *learning terms* Φ: these are the terms that drive the dynamics to align the student with the teacher, and consequently to reduce the population risk; it can be shown that these same terms are the only ones that survive in the *gradient flow* limit, and are therefore what drives the learning.

(ii) *variance term* Ψ: this is the term that arises from the stochasticity of the training process, and it is responsible for the fluctuations of the order parameters around their mean trajectory; notably, it is proportional to the learning rate γ and does *not* vanish for $\Delta = 0$. This is the crucial noise introduced by SGD, and is precisely what differentiates it from the gradient flow limit.

When label noise is present ($\Delta > 0$), the irreducible Δ-dependent contribution to the variance term is responsible for the *noise plateau* phenomenon [Vei+22]: the order parameters converge to a suboptimal fixed point and cannot reach the global minimum even as $t \to \infty$; see Figure 1.2.

Practitioners often address this problem with a decaying learning rate, which effectively makes the variance term vanish, combined with larger batch sizes, which reduce the variance of the stochastic

gradients by averaging out the label noise. It is natural to ask how Equations (SS-ODEs) change when the learning rate and the batch size are themselves allowed to scale with the dimension d; this is exactly the topic of Chapter 4. Letting both scale as powers of d modifies the Saad & Solla equations through *intra-batch*, finite-d corrections to the otherwise dimension-free dynamics, and uncovers a genuine *time/complexity trade-off*: a larger batch averages the gradient noise and suppresses the variance term Ψ, but beyond a critical size it no longer reduces the number of steps and merely consumes more samples per step. The resulting phase diagram, in the n_b–γ plane of the batch size and the learning rate, separates a regime in which the network fails to learn, an intermediate regime on the edge of learning where the plateau persists, and a regime of successful recovery, and it pinpoints the batch size that minimizes the training time.

1.3.4 Closed-form examples

Equation (SS-ODEs) can fully describe the dynamics only when it is actually possible to integrate analytically the expectations on the right-hand side, arriving at a closed-form expression for Φ and Ψ as functions of the order parameters. This depends strongly on the choice of the activation function σ and of the teacher function $h^\star$; in the literature the problem is often tackled in the simple setting of *matching architectures*.

Assume for simplicity that the *teacher* is itself a two-layer network, with k hidden neurons and the same activation function σ as the *student* (which has p hidden neurons):

$$f^\star(\boldsymbol{x}) = \frac{1}{k}\sum_{j=1}^{k} a_j^\star \sigma\left(\frac{\boldsymbol{w}_j^\star \cdot \boldsymbol{x}}{\sqrt{d}}\right) \quad \text{or equivalently} \quad h^\star(\boldsymbol{\lambda}^\star) = \frac{1}{k}\sum_{j=1}^{k} a_j^\star \sigma(\lambda_j^\star) \tag{1.16}$$

The integrability of Equation (SS-ODEs) then boils down to whether we can compute, for a generic covariance $C_p \in PSD\mathbb{R}^{p\times p}$, the following expectations:

$$I_2^{\text{risk}}(C_2) \coloneqq \mathbb{E}_{(z_1,z_2)\sim\mathcal{N}(0,C_2)}\left[\sigma(z_1)\sigma(z_2)\right], \tag{1.17}$$

$$I_2^{\text{noise}}(C_2) \coloneqq \mathbb{E}_{(z_1,z_2)\sim\mathcal{N}(0,C_2)}\left[\sigma'(z_1)\sigma'(z_2)\right], \tag{1.18}$$

$$I_3(C_3) \coloneqq \mathbb{E}_{(z_1,z_2,z_3)\sim\mathcal{N}(0,C_3)}\left[\sigma(z_1)\sigma'(z_2)z_3\right], \tag{1.19}$$

$$I_4(C_4) \coloneqq \mathbb{E}_{(z_1,z_2,z_3,z_4)\sim\mathcal{N}(0,C_4)}\left[\sigma'(z_1)\sigma'(z_2)\sigma(z_3)\sigma(z_4)\right]. \tag{1.20}$$

Here I_2^{risk} appears in the expression of the population risk and Φ^a, I_3 appears in the learning terms Φ^M and Φ^Q, while I_2^{noise} and I_4 appear in the variance term Ψ^Q.

A substantial body of literature has been devoted to whether these expectations admit a closed form, and hence to whether the asymptotic equations are exact, depending on the choice of the activation function σ.

- *linear* $\sigma(x) = x$: this case is trivial, since the student and the teacher are then just linear models, and the dynamics can be fully solved in closed form;
- *error function* $\sigma(x) = \text{erf}(x/\sqrt{2})$: this is the original setting of Saad and Solla [SS95a], where the equations can be fully solved in closed form and the dynamics fully characterized;

Some of these activations do not fully satisfy the assumptions of Theorem 1.1, yet they have been shown to work empirically; the theorem is, in all likelihood, not tight.

- *ReLU* $\sigma(x) = \max(0, x)$: this is the most common activation function used in practice, but it has no closed-form solution for the expectation I_4. The other integrals were studied in Yoshida and Okada [YO19] using results from Yoshida et al. [Yos+17];
- *phase retrieval* $\sigma(x) = x^2$: the study of Equations (SS-ODEs) for this activation function is the full content of Chapter 3. We determined the exact closed form of the expectations and used it to study the number of samples needed to learn the teacher network;
- *polynomial* $\sigma(x) = p(x)$: the framework introduced in Chapter 3 can be extended to any polynomial activation function. The code attached to that work can be used to compute the exact closed form of the expectations, although the number of terms grows exponentially with the degree of the polynomial, making it impractical for high-degree polynomials;
- *GELU* $\sigma(x) = x\Phi(x)$, where Φ denotes the standard Gaussian CDF: this is a smooth approximation of the ReLU function, and it has been shown to admit a closed-form solution for the expectations I_2 and I_3 [Ric+23].

The other activations commonly used in practice (such as sigmoid, tanh, and so on) do not admit a closed-form solution for the expectations, and the dynamics can then only be studied numerically, by approximating the expectations. One option is a Monte Carlo approximation of the expectations, but in our experience this is rarely precise enough and often leads to worse results than simulating SGD on the weights directly. A better option is to approximate the activation function with Hermite polynomials, for which the expectations do admit a closed form; a more advanced technique for approximating the expectations is discussed in Citton, Richert, and Biehl [CRB24], where the authors expand both the activation function and the correlated Gaussian measure in the Hermite basis to obtain a very precise approximation of the expectations for any activation function.

Expanding in Hermite polynomials along every dimension is essentially equivalent to a tensor-product Gauss–Hermite quadrature of the correlated expectation; this technique is used in Chapter 6, in a slightly different setting.

Even though the Saad & Solla equations are not always analytically solvable, they remain a very powerful tool for studying the dynamics of SGD around initialization: for small displacements, any σ and $h^\star$ can be expanded in a series of Hermite polynomials, which again yields a closed-form expression for the expectations and approximates the dynamics of the order parameters. This idea is at the core of several chapters of this thesis, and we develop it further in the next section.

1.3.5 Beyond the high-dimensional limit

When deriving Equations (SS-ODEs), we took the infinitesimal time step to be γ/pd, which indeed vanishes as $d \to \infty$. However, this is not the only way to obtain a vanishing time step:

- $\gamma \to 0$, d, p finite: this is the classical limit that, for online SGD, converges to *gradient flow* [RM51];
- $p \to \infty$, d, γ finite: this is the infinite-width limit, sometimes called the *mean-field* limit, studied for two-layer networks in [MMN18; CB18; RV22; SS20].

The key observation is that the convergence of the overlap process to these equations is, more generally, controlled by the effective time step γ/pd alone: a non-asymptotic refinement of Theorem 1.1 due to Veiga et al. [Vei+22] bounds the deviation from the deterministic trajectory by a

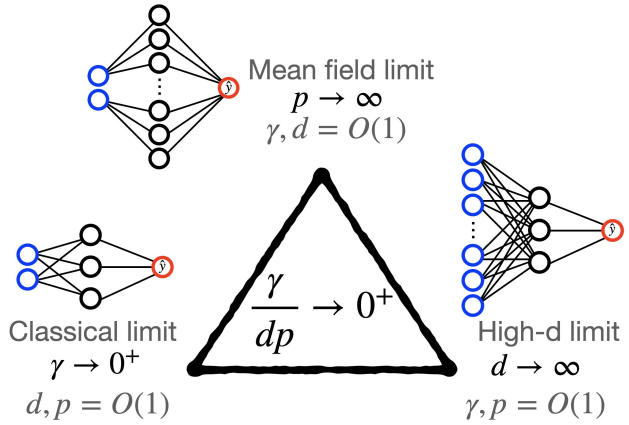


Figure 1.3: The three limiting regimes of one-pass SGD and the regimes that interpolate them. As long as the effective time step satisfies $\gamma/pd \to 0^+$, the order-parameter dynamics admits a deterministic, low-dimensional description. The corners are the high-dimensional ($d \to \infty$), classical gradient-flow ($\gamma \to 0$), and mean-field ($p \to \infty$) limits, while the edges and interior interpolate between them; see Chapter 2 for details.

quantity that vanishes whenever $\gamma/pd \to 0^+$, regardless of how that limit is reached. The deterministic, low-dimensional description in terms of the order parameters is therefore valid *beyond* the high-dimensional limit on which the original Saad & Solla analysis focused, and the three limits above can be treated on the same footing. This equivalence holds, however, only on finite time horizons $t = \mathcal{O}(\log d)$: the bound of Veiga et al. [Vei+22] carries a prefactor that grows exponentially in time, so it controls the $\mathcal{O}(1)$-time regimes treated here but not the much longer timescales on which learning escapes its initial plateau. Those long-time results rest on separate stochastic arguments rather than on the deterministic ODE, as discussed in Section 1.4 and quantified in Appendix A.2.

In Chapter 2 we build on this observation to study the connection between the Saad & Solla equations and the other two limits within a single framework, summarized by the phase diagram of Figure 1.3. Its three corners are the high-dimensional, classical gradient-flow, and mean-field limits, while the edges and interior cover combinations such as the *high-dimensional gradient-flow* and *high-dimensional mean-field* limits, which are not captured by the classical Saad & Solla equations.

The mean-field corner is the most delicate. Because Equations (SS-ODEs) track the $O(p^2 + pk)$ order parameters, they become intractable precisely in the overparametrized regime $p \to \infty$ for which the mean-field description [MMN18; CB18; RV22; SS20] was devised. We bridge the two by exploiting the fact that, throughout training, the student weights remain close to the low-dimensional subspace spanned by the teacher directions: projecting the dynamics onto this subspace collapses the growing system of ODEs back to a description of constant dimension, recovering a mean-field PDE while keeping the explicit dependence on d. This connects the order-parameter picture of this section to the hydrodynamic, optimal-transport viewpoint of the mean-field literature.

1.4 Information exponent and beyond

The core question of this section is to characterize the sample complexity of learning a function $h^\star$, given an expressive enough student. We show that the answer is controlled by a single integer, the *information exponent* of Ben Arous, Gheissari, and Jagannath [BGJ21], which measures the flatness of the population risk at the initialization saddle. We then discuss two generalizations: the *leap exponent* of Abbe, Adsera, and Misiakiewicz [AAM22] and Abbe, Adserà, and Misiakiewicz [AAM23], which governs the multi-index case $k > 1$, and the *generative exponent* of Damian et al. [Dam+24], which measures the hardness of the problem under a broader class of algorithms.

We focus for now on the *single-index* setting, where the teacher is a single-index model $f^\star(\boldsymbol{x}) = h^\star(\boldsymbol{w}^\star \cdot \boldsymbol{x}/\sqrt{d})$ and the student has the same shape $f(\boldsymbol{x}) = \sigma(\boldsymbol{w} \cdot \boldsymbol{x}/\sqrt{d})$ with a different activation function σ. The overlap matrices of Equation (1.8) then collapse to two scalars, the *self-overlap* and the *teacher overlap*,

$$q = \frac{\|\boldsymbol{w}\|^2}{d}, \qquad m = \frac{\langle \boldsymbol{w}, \boldsymbol{w}^\star \rangle}{d}, \tag{1.21}$$

and the pre-activations are jointly Gaussian, $(\lambda, \lambda^\star) \sim \mathcal{N}(0, \Omega)$ with covariance

$$\Omega = \left(\begin{smallmatrix} q & m \\ m & 1 \end{smallmatrix}\right).$$

We make one more assumption on top of the setting of Section 1.2:

Assumption 1.1 (Projected SGD)**.** *The student is trained with projected SGD, where each step is projected back onto the sphere $\mathbb{S}^{d-1}(\sqrt{d})$; in other words, the update rule*

$$\boldsymbol{w}^{\nu+1} = \sqrt{d}\,\frac{\boldsymbol{w}^\nu - \gamma \nabla_{\boldsymbol{w}} \ell(f(\boldsymbol{x}^\nu), y^\nu)}{\left\| \boldsymbol{w}^\nu - \gamma \nabla_{\boldsymbol{w}} \ell(f(\boldsymbol{x}^\nu), y^\nu) \right\|}.$$

The effect of this is to keep the norm of the student weights fixed, $q = 1$, so that the only degree of freedom is the teacher overlap m. This technical simplification lets us focus on the true signal, the correlation with the teacher direction, by decoupling it from changes in the student-weight norm, which can alter the risk without contributing to recovery; see Chapter 3 for a detailed discussion. The Saad & Solla equations (SS-ODEs) then reduce to a single ODE for m that naturally encodes the spherical constraint

We are actually neglecting the *variance term* Ψ in the self-overlap equation, which is subleading in the small-learning-rate regime that governs the escape from initialization; in Chapter 3 we keep it and show that the derivation goes through.

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \underbrace{\mathbb{E}\Big[\big(h^\star(\lambda^\star) - \sigma(\lambda)\big)\,\sigma'(\lambda)\,\lambda^\star\Big]}_{\text{drift}} - m\,\underbrace{\mathbb{E}\Big[\big(h^\star(\lambda^\star) - \sigma(\lambda)\big)\,\sigma'(\lambda)\,\lambda\Big]}_{\text{spherical constraint}} \tag{1.22}$$

The equation above can be rearranged in a more compact and insightful form

$$\frac{\mathrm{d}m}{\mathrm{d}t} = (1 - m^2)\,C'(m) \quad \text{with } C(m) = \mathbb{E}[h^\star(\lambda^\star)\,\sigma(\lambda)]. \tag{1.23}$$

The derivation is reported in Appendix A.1. The prefactor $(1 - m^2)$ is purely *geometric*: the spherical Jacobian $\sin^2\theta$ for $m = \cos\theta$. Hence the activations σ and $h^\star$ enter the dynamics only

through $C(m)$, and the equation is exact for every $m \in [-1, 1]$, not merely near initialization. On the sphere the marginal terms of $\mathcal{R}$ are frozen, leaving

$$\mathcal{R}(m) = \text{const} - C(m), \tag{1.24}$$

so that the overlap dynamics is gradient ascent on C.

Since one-pass SGD consumes one sample per gradient step, the sample complexity equals

$$n = \frac{d}{\gamma}\, t,$$

so the time t serves as a direct proxy for the number of samples up to the constant d/γ. Although the ODE (1.23) is dimension-independent, its initial condition is not: at random initialization $\boldsymbol{w}^0 \sim \mathcal{N}(0, I_d)$, the student-teacher overlap is a sum of d independent mean-zero terms, and is therefore tiny,

$$m^0 = \frac{\langle \boldsymbol{w}^0, \boldsymbol{w}^\star \rangle}{d} = \mathcal{O}_d\!\left(\frac{1}{\sqrt{d}}\right).$$

The student starts nearly orthogonal to the teacher, and the dynamics must drag m from $\mathcal{O}(1/\sqrt{d})$ up to a finite, d-independent value before any meaningful alignment is established. This is the bottleneck that determines the d-scaling of the total sample count; once m reaches a finite value, the remaining optimization is fast and contributes only sub-leading terms. The relevant milestone is therefore *weak recovery*.

Definition 1.1 (Weak recovery)**.** *We say that the student has achieved weak recovery of the teacher with threshold η if the teacher overlap m has reached a finite value η independent of the dimension d:*

$$m = \frac{\langle \boldsymbol{w}, \boldsymbol{w}^\star \rangle}{d} \geq \eta, \qquad \eta \in (0, 1). \tag{1.25}$$

We denote by $t_{weak}(\eta)$ the first time at which the solution to Equation (1.23) *reaches $m(t_{weak}) = \eta$.*

The sample complexity of weak recovery is therefore $n = \mathcal{O}_d(d\, t_{\text{weak}}(\eta))$.

In practice, we can take η small enough that Equation (1.23) is well approximated by its leading order expansion around $m = 0$

$$\frac{\mathrm{d}m}{\mathrm{d}t} \propto C^{(\ell)}(0)\, m^{\ell-1}, \qquad \ell = \min\{j \geq 1 : C^{(j)}(0) \neq 0\}. \tag{1.26}$$

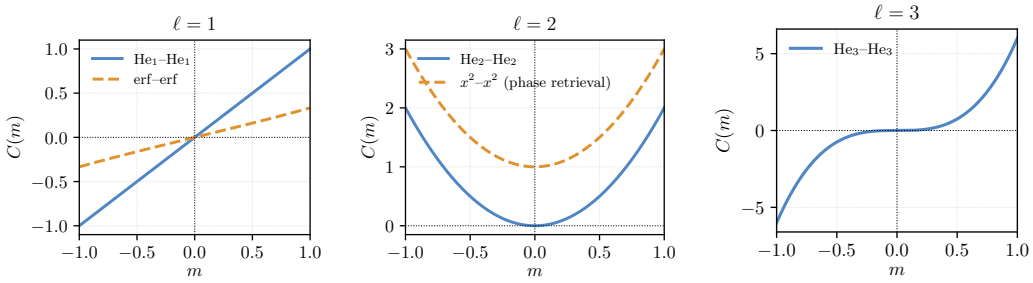
The actual factor in front of $m^{\ell-1}$ is not dependent on d and does not affect the final escaping complexity. In the same way, the factor $(1 - m^2)$ does not change the leading order of the expansion.

We can better understand the meaning of this by looking at some examples.

Example 1.1 (Linear regression)**.** *For the linear regression target $h^\star(\lambda^\star) = \lambda^\star$ and the linear activation $\sigma(\lambda) = \lambda$, the correlation potential is $C(m) = m$, so $C'(m) = 1$ and*

$$\frac{\mathrm{d}m}{\mathrm{d}t} = 1 - m^2 \overset{m \ll 1}{\approx} 1 \quad \textit{with i.c. } m^0 \approx 1/\sqrt{d}. \tag{1.27}$$

The solution is $m(t) = m^0 + t$, so the weak recovery time is $t_{weak}(\eta) \approx \eta$, giving a total sample complexity of $n = \mathcal{O}_d(d)$.



(a) $\ell = 1$ (b) $\ell = 2$ (c) $\ell = 3$

Figure 1.4: **Correlation potential $C(m)$ for three values of the information exponent.** Each panel plots $C(m) = \mathbb{E}[h^\star(\lambda^\star)\sigma(\lambda)]$ as a function of the student–teacher overlap $m \in [-1, 1]$. *(a)* $\ell = 1$: both He_1–He_1 (linear regression) and erf–erf have a non-zero slope at $m = 0$, so gradient ascent on C immediately pushes the student away from mediocrity. *(b)* $\ell = 2$: He_2–He_2 and the phase-retrieval target x^2–x^2 differ by a constant, but since the ODE depends only on $C'(m)$ the dynamics is identical; both are flat to first order at $m = 0$. *(c)* $\ell = 3$: the potential is cubic, flat to second order at $m = 0$, causing the longest search phase and hence the largest sample complexity $n = \mathcal{O}(d^{\ell-1})$.

Example 1.2 (Phase retrieval)**.** *For the phase retrieval target $h^\star(\lambda^\star) = (\lambda^\star)^2$ and the quadratic activation $\sigma(\lambda) = (\lambda)^2$, the correlation potential is $C(m) = 1 + 2m^2$, so $C'(m) = 4m$ and*

$$\frac{\mathrm{d}m}{\mathrm{d}t} = 4\,m\,(1 - m^2) \overset{m \ll 1}{\approx} 4\,m \quad \textit{with i.c. } m^0 \approx 1/\sqrt{d}. \tag{1.28}$$

The solution is $m(t) = m^0\, e^{4t}$, so the weak recovery time is $t_{\mathit{weak}}(\eta) \approx \frac{1}{4}\log(\eta\sqrt{d})$, giving a total sample complexity of $n = \mathcal{O}_d(d \log d)$. In Chapter 3 we carry out a finer analysis of the equation, studying the precise constants and how they scale with the learning rate γ and the label noise Δ.

The predicted $t_{\mathit{weak}}(\eta) = \mathcal{O}(\log d)$ is exactly the time scale of the validity window of Theorem 1.1; this passage requires a more careful analysis, see Appendix A.2.

In summary: the flatter the correlation potential (equivalently, the population risk) at initialization, the harder the escape from the saddle at $m = 0$, and the more samples needed to achieve weak recovery. Geometrically, in weight space, the population risk at $\boldsymbol{w}^0$ is a *strict saddle*, flat along all but one direction—a golf course with a single hole—so the student is stuck at *mediocrity*, only weakly correlated with the teacher. In the space of the order parameter m, this translates into a flat potential $C(m)$ at $m = 0$ (Figure 1.4), with initialization very close to the critical point.

The integer ℓ in Equation (1.26) is the *information exponent* of Ben Arous, Gheissari, and Jagannath [BGJ21], and it is the only quantity that matters for the sample complexity of learning a single-index target with SGD. Expanding $h^\star = \sum_k a_k \mathrm{He}_k$ and $\sigma = \sum_k b_k \mathrm{He}_k$, and using the orthogonality of Hermite polynomials under a correlated Gaussian, $\mathbb{E}[\mathrm{He}_j(\lambda)\mathrm{He}_k(\lambda^\star)] = \delta_{jk}\, k!\, m^k$; the correlation is then a pure power series in the overlap,

$$C(m) = \sum_{k \geq 0} k!\, a_k\, b_k\, m^k, \tag{1.29}$$

so that near the saddle $\mathcal{R}(m) - \mathcal{R}(0) = -\sum_{k \geq 1} k!\, a_k b_k\, m^k$.

Assumption 1.2 (σ excites the ℓ-th mode)**.** *The student activation σ is such that*

$$b_\ell = \frac{1}{\ell!}\,\mathbb{E}[\sigma\,\mathrm{He}_\ell] \neq 0.$$

In other words, the student is expressive enough to excite the mode of any teacher.

The first term that bends the landscape is the lowest k with $a_k \neq 0$—which is exactly ℓ. In words, the information exponent is the order to which the population risk is flat at the mediocrity saddle: $\mathcal{R}(m) - \mathcal{R}(0) \approx -\,\ell!\,a_\ell b_\ell\,m^\ell$. Near the saddle the geometric factor in (1.23) is $1 - m^2 \approx 1$, so the escape drift inherits the same order,

$$\frac{\mathrm{d}m}{\mathrm{d}t} = (1-m^2)\,C'(m) \;\sim\; \ell\,\ell!\,a_\ell b_\ell\,m^{\ell-1}. \tag{1.30}$$

We can now state the original definition of the information exponent.

Definition 1.2 (Information exponent [BGJ21])**.** *The information exponent ℓ of a single-index target $h^\star$ is the lowest order of the Hermite expansion of $h^\star$ with non-zero coefficient,*

$$\ell = \min\{j \geq 1 : a_j = \frac{1}{j!}\,\mathbb{E}_{z\sim\mathcal{N}(0,1)}[h^\star(z)\,\mathrm{He}_j(z)] \neq 0\}. \tag{1.31}$$

This Hermite-degree characterization has an early precursor in the Gaussian-space analysis of single- and multi-index models by Dudeja and Hsu [DH18], who showed that for Gaussian covariates the population landscape of the maximum-likelihood objective is free of spurious minima and gave an iterative procedure recovering the index space with a sample complexity governed by the degree of its first informative Hermite component.

The last piece of the puzzle is to understand how the information exponent controls the sample complexity of learning. Integrating the deterministic drift (1.30) from $m^0 = \mathcal{O}(1/\sqrt{d})$ gives the escape time, hence the number of samples:

$$t_{\text{escape}} \asymp \begin{cases} 1 & \ell = 1, \\ \log d & \ell = 2, \\ d^{\ell/2-1} & \ell \geq 3. \end{cases} \tag{1.32}$$

However, the escape time predicted by Equation (1.30) is not sufficient to determine the sample complexity for $\ell \geq 3$. The obstruction is that Theorem 1.1 certifies the ODE as a faithful proxy for the SGD process only over a *finite* time horizon. The theorem itself remains *true* for every ℓ, but for $\ell \geq 3$ the escape occurs at $t_{\text{escape}} \gg \log d$, far outside this window: there the deterministic ODE is simply no longer close to the SGD trajectory, so the exit time it predicts carries no meaning. The deterministic drift captured by the ODE is only the mean of the stochastic dynamics, and the potential is so flat that the fluctuations of the order parameter around the mean trajectory cannot be neglected. Effectively, the dynamics can be divided in two parts:

The convergence guarantee degrades as $e^{Ct}/\sqrt{d}$, the sharp form of the bound is derived in Appendix A.2. Even for $\ell = 2$ the stochastic noise already dominates the escape, yet the drift still fixes the correct scaling.

- *search phase*: the student is stuck in the flat region around $m = 0$, and the dynamics is dominated by the stochastic fluctuations of the order parameter; the escape time is therefore longer than predicted by the deterministic ODE (1.23) for $\ell \geq 3$;

- *descent phase*: once the student has escaped the flat region, the drift dominates and the dynamics is well approximated by the ODE (1.23); the duration of this phase is therefore well captured by the Saad & Solla equations (SS-ODEs).

The *search phase* is the bottleneck that determines the scaling of the sample complexity with d, while the *descent phase* always happens linearly in d and contributes to the sample complexity only through sub-leading terms, for $\ell \geq 2$. Capturing the search phase therefore demands tools that go beyond the deterministic ODE. Ben Arous, Gheissari, and Jagannath [BGJ21] track the overlap directly through its noisy stochastic recursion, controlling it with a *martingale decomposition* of the summary statistic together with a hitting-time analysis of the escape from the saddle; they explicitly bypass the classical ODE method, which is blind to the search phase precisely because there the martingale fluctuations outweigh the increments of the drift. This yields the following theorem.

Every information exponent $\ell \geq 2$ hides an extra polylogarithmic factor, present but not proven sharp. Informally we quote the sharp bound; the formal proofs keep all logarithmic factors, a convention we follow throughout the thesis.

Theorem 1.2 (Informal version of Ben Arous, Gheissari, and Jagannath [BGJ21])**.** *The sample complexity of learning a single-index target with information exponent ℓ with one-pass SGD is*

$$n = \begin{cases} \mathcal{O}(d) & \ell = 1, \\ \mathcal{O}(d \log d) & \ell = 2, \\ \mathcal{O}(d^{\ell-1}) & \ell \geq 3. \end{cases} \tag{1.33}$$

The concept of information exponent is actually more general than the one presented here. Every problem whose loss depends only on the scalar product $\langle \boldsymbol{w}, \boldsymbol{x} \rangle$ and is optimized by one-pass spherical SGD has a well-defined information exponent that controls the sample complexity of learning. This includes, for example, online PCA, spiked matrix/tensor estimation, or Gaussian mixture clustering, as in the seminal work of Ben Arous, Gheissari, and Jagannath [BGJ21].

Chapter 4 extends the exponent to the single-index case where n_b and γ scale with d. The chapter reveals a sharp threshold at $n_b \sim d^{\ell/2}$: below it, scaling the learning rate proportionally to the batch size reduces the number of gradient steps proportionally while keeping the total sample complexity $N = n_b T = \mathcal{O}(d^{\ell-1})$ unchanged, offering a practical speed-up through parallelism. Above the threshold, continuing to scale the learning rate this way makes the gradient signal dominated by self-interaction terms that contract the correlation to zero, and standard SGD fails to achieve weak recovery; holding the learning rate fixed at $\gamma = \mathcal{O}(1)$ instead, SGD still weakly recovers the target, but the number of steps stops decreasing with n_b and the sample complexity grows to $N = \mathcal{O}(n_b d^{\ell/2-1})$, so the extra samples per step are wasted rather than traded for speed. A modified algorithm, Correlation Loss SGD, suppresses this self-interaction term and restores the optimal sample complexity $N = \mathcal{O}(d^{\ell-1})$ up to $n_b \sim d^{\ell-1}$. Figure 1.5 shows the resulting phase diagram.

Chapter 6 extends the exponent to a model that proxies the attention layer. The input $X \in \mathbb{R}^{L \times d}$ is now a sequence of L d-dimensional vectors, and our learned model is

The bilinear read-out is displayed here for concreteness. Chapter 6 establishes its results for a general *reduction map* $\Pi : \mathbb{R}^{L \times L} \to \mathbb{R}^{k_{\text{out}}}$, where k_{out} is the attention output dimension, distinct from the teacher-direction count k.

$$f(X) = \boldsymbol{a}_L^\top \operatorname{softmax}\left(\left(X + \frac{P}{\sqrt{d}} \right) \boldsymbol{w}\boldsymbol{w}^\top \left(X + \frac{P}{\sqrt{d}} \right)^\top \right) \boldsymbol{a}_R,$$

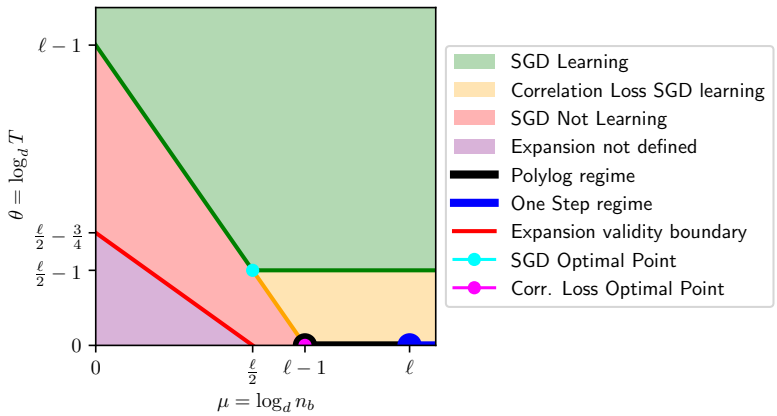


Figure 1.5: **Time / Batch size trade-off for weak recovery** (from Chapter 4). Phase diagram of the weak-recovery time exponent $\theta = \log_d T$ as a function of the batch-size exponent $\mu = \log_d n_b$, for a target with information exponent $\ell \geq 3$. At $\mu = 0$ (one sample per step) the time exponent is $\ell - 1$, the top of the accessible region. Increasing n_b up to the optimal point $n_b \sim d^{\ell/2}$ trades iterations for parallelism at constant total sample cost $N = \mathcal{O}(d^{\ell-1})$. Beyond this, standard SGD's optimal time plateaus at $T = \mathcal{O}(d^{\ell/2-1})$ rather than continuing to fall, so the extra samples per step are wasted; pushing the learning rate to shrink T further enters the self-interaction regime and not correlating region, where standard SGD cannot achieve weak recovery at all. Correlation Loss SGD extends the accessible region to its optimal point at $n_b \sim d^{\ell-1}$.

where P is a fixed positional encoding, and $\boldsymbol{a}_L, \boldsymbol{a}_R \in \mathbb{R}^L$ are the read-out vectors. The information exponent is then defined as the smallest degree of the Hermite multivariate expansion of the target function, now called the *Sequence Information Exponent* (SIE), and the sample complexity obeys the same d-scaling hierarchy as (1.33) with ℓ replaced by the SIE. The order-parameter space expands from the scalar m to the pair $(\boldsymbol{e}, m)$, where $m = \boldsymbol{w}^\star \cdot \boldsymbol{w}/d$ is the teacher overlap and $\boldsymbol{e} = P\boldsymbol{w}/\sqrt{d}$ encodes the alignment of the student weights with the positional encoding P; the dynamics is described by a coupled ODE in $(\boldsymbol{e}, m)$, illustrated in Figure 1.6.

1.4.1 Leap exponent

When the teacher has more than one relevant direction ($k > 1$), the single-index picture extends naturally. The teacher is now a *multi-index model*,

$$f^\star(\boldsymbol{x}) = h^\star\left(\frac{W^\star \boldsymbol{x}}{\sqrt{d}}\right), \qquad W^\star = (\boldsymbol{w}_1^\star, \dots, \boldsymbol{w}_k^\star)^\top \in \mathbb{R}^{k\times d}, \tag{1.34}$$

and the single overlap m of Equation (1.21) broadens to an overlap vector $\boldsymbol{m} \in \mathbb{R}^k$,

$$m_r = \frac{\langle \boldsymbol{w}, \boldsymbol{w}_r^\star \rangle}{d}, \qquad r \in [k]. \tag{1.35}$$

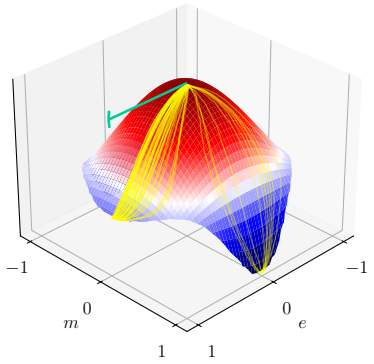


Figure 1.6: **Population loss of the Sequence Single Index Model in the** (e, m) **plane** (from Chapter 6). Working with two order parameters instead of one can produce richer loss landscapes, including multiple local minima. Here the semantic minimum at $(e, m) = (0, 1)$ is the global one, yet SGD trajectories (yellow) are attracted to the local positional minimum at $(e, m) = (1, 0)$.

The multivariate target $h^\star : \mathbb{R}^k \to \mathbb{R}$ admits a Hermite expansion in the teacher pre-activations $\boldsymbol{\lambda}^\star = W^\star \boldsymbol{x}/\sqrt{d}$,

$$h^\star(\boldsymbol{\lambda}^\star) = \sum_{\alpha \in \mathbb{N}^k} a_\alpha \prod_{r=1}^{k} \mathrm{He}_{\alpha_r}(\lambda_r^\star), \qquad a_\alpha = \frac{1}{\alpha!}\,\mathbb{E}\left[h^\star(\boldsymbol{\lambda}^\star) \prod_{r=1}^{k} \mathrm{He}_{\alpha_r}(\lambda_r^\star)\right]. \tag{1.36}$$

Lifting the degree ℓ part to the ambient space $\mathbb{R}^d$ gives the *Hermite tensor* of order ℓ,

$$T_\ell = \sum_{\substack{\alpha \in \mathbb{N}^k \\ |\alpha| = \ell}} a_\alpha \,\mathrm{sym}\Big(\boldsymbol{w}_1^{\star \otimes \alpha_1} \otimes \cdots \otimes \boldsymbol{w}_k^{\star \otimes \alpha_k}\Big) \in (\mathbb{R}^d)^{\otimes \ell}. \tag{1.37}$$

Exactly as in the single-index case (1.30), the escape from initialization ($\boldsymbol{m} \approx 0$) is governed by the leading non-trivial tensor. The population gradient drives the student weights along the contraction of T_ℓ with $\boldsymbol{w}$ itself,

$$\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}t} \;\propto\; \ell!\, b_\ell\, T_\ell[\boldsymbol{w}^{\otimes(\ell-1)}], \qquad \ell = \min\{|\alpha| : a_\alpha \neq 0\}, \tag{1.38}$$

and the overlap vector follows by projection onto the teacher directions,

$$\frac{\mathrm{d}m_r}{\mathrm{d}t} = \frac{1}{d}\left\langle \boldsymbol{w}_r^\star,\, \frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}t} \right\rangle, \qquad r \in [k]. \tag{1.39}$$

Here b_ℓ is the ℓ-th Hermite coefficient of σ, and $T_\ell[\boldsymbol{w}^{\otimes(\ell-1)}] \in \mathbb{R}^d$ denotes the contraction of T_ℓ with $(\ell - 1)$ copies of $\boldsymbol{w}$; each such contraction brings down a factor $\langle \boldsymbol{w}_s^\star, \boldsymbol{w} \rangle = d\, m_s$, so the drift of $\boldsymbol{m}$ is a polynomial of degree $\ell - 1$ in the overlaps—the multi-index counterpart of

the $m^{\ell-1}$ drift of (1.30). The integer $\ell = \min\{|\alpha| : a_\alpha \neq 0\}$ is the multi-index analogue of the information exponent, the lowest degree of the non-trivial Hermite tensors of $f^\star$, and, exactly as in the single-index case, it fixes the cost $\mathcal{O}(d^{\ell-1})$ of the *first* escape from the flat saddle.

An equivalent formulation, direction by direction, makes the analogy with the single-index case fully explicit. For each direction $\boldsymbol{v}$ in the teacher subspace $V^\star = \mathrm{span}(\boldsymbol{w}_1^\star, \dots, \boldsymbol{w}_k^\star)$, define the *directional information exponent* (see Chapter 5) as

$$\ell_{\boldsymbol{v}} = \min\left\{ j \geq 1 : \mathbb{E}\Big[y \, \mathrm{He}_j\big(\langle \boldsymbol{v}, \boldsymbol{x} \rangle\big)\Big] \neq 0 \right\}; \tag{1.40}$$

then $\ell = \min_{\boldsymbol{v} \in V^\star} \ell_{\boldsymbol{v}}$, and the direction achieving the minimum is the one the student aligns with first. When a single escape suffices, this exponent already coincides with the *leap complexity* of Abbe, Adsera, and Misiakiewicz [AAM22] and Abbe, Adserà, and Misiakiewicz [AAM23]; in general, as we see next, the leap can exceed ℓ.

Staircase dynamics Capturing the staircase dynamics requires the student to also have $p \geq k$ neurons,

$$f(\boldsymbol{x}) = \frac{1}{p} \sum_{j=1}^{p} a_j \, \sigma\left(\frac{\boldsymbol{w}_j \cdot \boldsymbol{x}}{\sqrt{d}} \right), \tag{1.41}$$

each with its own overlap vector $\boldsymbol{m}_j = W^\star \boldsymbol{w}_j / d \in \mathbb{R}^k$. Every neuron starts from an independent random initialization and evolves according to Equation (1.38), so different neurons can specialize to different teacher directions.

Multi-index learning unfolds as a *staircase* [AAM22; AAM23; Dan+24a]. At initialization all overlaps are $\mathcal{O}(1/\sqrt{d})$, so all terms in the expansion (1.36) vanish and the dynamics is flat in every direction. Neurons first align with the direction $\boldsymbol{v}$ achieving the minimum $\ell_{\boldsymbol{v}} = \ell$, requiring $\mathcal{O}(d^{\ell-1})$ samples. Once that direction is weakly recovered, its overlap m_r reaches an $\mathcal{O}(1)$ value and henceforth acts as a constant multiplier: every coefficient a_α with $\alpha_r > 0$ in (1.36) now contributes a finite term to the correlation rather than vanishing. This lowers the effective directional information exponent of the remaining directions: the exponent that was $\ell_{\boldsymbol{v}'}$ (minimum total degree with $\alpha_{v'} > 0$) becomes the minimum degree in direction $\boldsymbol{v}'$ alone, which can be strictly smaller, and the next escape costs fewer samples. The dynamics progresses from saddle to saddle in turn.

Example 1.3 (The staircase $\lambda_1^\star + \lambda_1^\star\lambda_2^\star + \lambda_1^\star\lambda_2^\star\lambda_3^\star$)**.** *Consider the $k = 3$ target*

$$h^\star(\boldsymbol{\lambda}^\star) = \lambda_1^\star + \lambda_1^\star\lambda_2^\star + \lambda_1^\star\lambda_2^\star\lambda_3^\star.$$

Its only nonzero coefficients in (1.36) *sit at*

$$\alpha \in \{(1,0,0),\ (1,1,0),\ (1,1,1)\}.$$

Here $\ell = 1$, set by the linear rung $\lambda_1^\star$, and since every rung adds a single new direction the leap is Leap $= 1$ *as well; the learning happens in 3 steps:*

(i) the dynamics escapes the flat saddle along $\boldsymbol{w}_1^\star$ in $\mathcal{O}(d)$ samples;

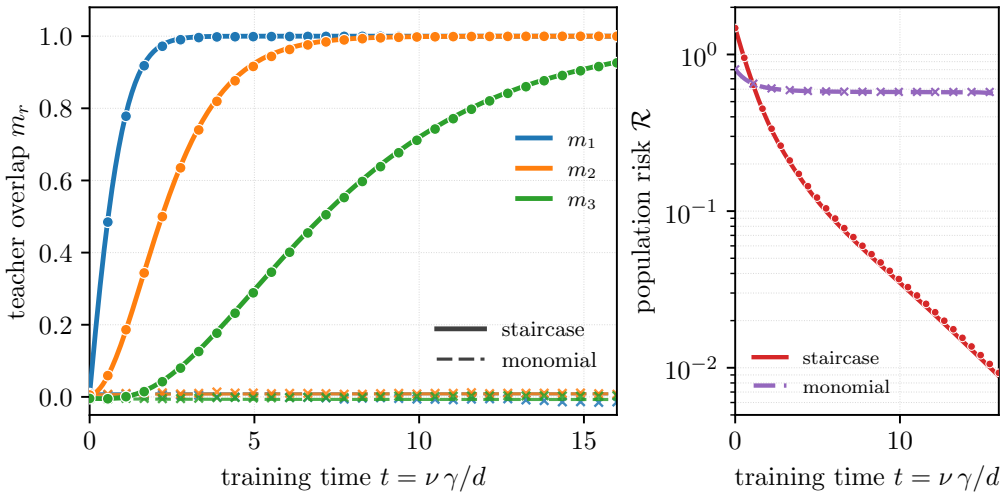


Figure 1.7: Learning the staircase $\lambda_1^\star + \lambda_1^\star\lambda_2^\star + \lambda_1^\star\lambda_2^\star\lambda_3^\star$ with a matched student architecture: online SGD (dots) matches the corresponding Saad & Solla equations (lines). *Left*, overlaps m_r; *right*, population risk. The staircase (solid) is an effective staircase—learned one direction after another (m_1, then m_2, then m_3); the bare monomial $\lambda_1^\star\lambda_2^\star\lambda_3^\star$ (dashed) is too hard and learns nothing.

(ii) once $m_1 = \mathcal{O}(1)$, the quadratic rung reads as $\lambda_1^\star\lambda_2^\star \approx m_1\,\lambda_2^\star$: a linear *signal in the second direction, whose effective degree has dropped to one. So $\boldsymbol{w}_2^\star$ is now leap-one and m_2 escapes in a further $\mathcal{O}(d)$;*

(iii) with $m_1, m_2 = \mathcal{O}(1)$, the cubic rung reads as $\lambda_1^\star\lambda_2^\star\lambda_3^\star \approx m_1 m_2\,\lambda_3^\star$ and m_3 follows in a final $\mathcal{O}(d)$. The three phases climb the staircase in $\mathcal{O}(d)$ samples in total.

The leap stays at one even though the target has degree three: each overlap, once recovered, lowers the effective degree of the rung above it to one. The intermediate rungs are precisely what make the climb cheap: strip them away and the bare top monomial $\lambda_1^\star\lambda_2^\star\lambda_3^\star$ has leap three, costing $\mathcal{O}(d^2)$. Figure 1.7 illustrates both regimes numerically.

The overall cost is therefore set not by the first escape but by the *hardest* step along the way. Following Abbe, Adserà, and Misiakiewicz [AAM23], the *leap* is the largest number of new teacher directions any single step must recover at once, minimized over the order in which the directions are learned, equivalently, the maximal effective directional exponent met along the staircase. Formally, for a set S of already-recovered directions, the effective exponent of a remaining $\boldsymbol{v}$ is the degree carried by the not-yet-recovered coordinates of the lowest monomial involving $\boldsymbol{v}$,

$$\ell_{\boldsymbol{v}}^{S} = \min\Big\{ \textstyle\sum_{r:\,\boldsymbol{w}_r^\star \notin S} \alpha_r \;:\; a_\alpha \neq 0,\; \alpha_{\boldsymbol{v}} > 0 \Big\}, \tag{1.42}$$

and the leap is the smallest achievable value of the worst step,

$$\text{Leap} = \min_{\pi\in\mathfrak{S}_k} \max_{r\in[k]} \ell_{\boldsymbol{w}_{\pi(r)}^\star}^{S_{r-1}}, \qquad S_{r-1} = \{\boldsymbol{w}_{\pi(1)}^\star, \dots, \boldsymbol{w}_{\pi(r-1)}^\star\}. \tag{1.43}$$

It is the leap, not the first-escape exponent ℓ, that fixes the total sample complexity $\mathcal{O}(d^{\text{Leap}-1})$: targets whose directions are all linearly connected (leap one) are recovered in $\mathcal{O}(d)$ samples, while each additional unit of leap raises the cost in direct analogy with (1.32).

Example 1.4 (A disconnected leap: $\lambda_1^\star + \lambda_2^\star\lambda_3^\star\lambda_4^\star$)**.** *The rung $\lambda_1^\star$ has directional exponent* 1*, while the block $\lambda_2^\star\lambda_3^\star\lambda_4^\star$, sharing no coordinate with it, has exponent* 3*. The leap is the maximum,* $\text{Leap} = 3$*, so the cost is $\mathcal{O}(d^2)$ even though* $\min_{\boldsymbol{v}} \ell_{\boldsymbol{v}} = 1$.

Example 1.5 (A nested leap: $\lambda_1^\star + \lambda_1^\star\lambda_2^\star\lambda_3^\star\lambda_4^\star$)**.** *The first rung again has exponent* 1*, but conditioned on recovering $\boldsymbol{w}_1^\star$ the block reduces to $m_1\,\lambda_2^\star\lambda_3^\star\lambda_4^\star$, still three new directions at once, effective exponent* 3*. So again* $\text{Leap} = 3$ *and the cost is $\mathcal{O}(d^2)$, despite the monomial $\lambda_1^\star\lambda_2^\star\lambda_3^\star\lambda_4^\star$ having* $\text{Leap} = 4$ *in isolation.*

1.4.2 Generative exponent

Returning to the single-index setting, the population risk on the sphere is $\mathcal{R}(m) = \text{const} - C(m)$ (Equation (1.24)). The gradient with respect to the student weights is therefore

$$\nabla_{\boldsymbol{w}}\mathcal{R} = -\,\mathbb{E}\left[y\,\sigma'(\lambda)\,\frac{\boldsymbol{x}}{\sqrt{d}}\right],$$

which is a *correlational query*: a functional of the input that is *linear* in the label y. This is the fundamental constraint of one-pass SGD: each gradient step can only ask "how correlated is y with this function of $\boldsymbol{x}$?". It is shared by every algorithm that processes each sample exactly once.

The class of algorithms restricted to queries of the form $\mathbb{E}[y\,\varphi(\boldsymbol{x})]$ for arbitrary $\varphi : \mathbb{R}^d \to \mathbb{R}$ is called the *Correlated Statistical Query* (CSQ) model. Damian et al. [Dam+23] established a CSQ lower bound: no CSQ algorithm can achieve weak recovery of a target with information exponent ℓ with fewer than $d^{\ell/2}$ samples. This bound is strictly tighter than the $d^{\ell-1}$ rate of one-pass SGD for $\ell \geq 3$; the gap between the two arises from the stochastic fluctuations that dominate the escape from the flat saddle, which loss smoothing suppresses. The CSQ bound $d^{\ell/2}$ is tight: the *partial-trace* algorithm of Anandkumar et al. [Ana+17] achieves it by directly estimating the population Hermite tensor T_ℓ via a spectral method (see Appendix A.3); Damian et al. [Dam+23] propose a gradient-based algorithm on a smoothed loss that achieves the same bound. However, it is still a CSQ algorithm: it correlates the raw label y linearly with a function of $\boldsymbol{x}$, and therefore cannot break the CSQ barrier.

Both one-pass SGD and the partial-trace algorithm are restricted to querying the label in its raw form, without prior nonlinear transformation. The key question is: what if we first apply a nonlinear transformation g to the label, and then query $\mathbb{E}[g(y)\,\varphi(\boldsymbol{x})]$? The minimum Hermite degree for which such a query can be non-zero defines a new exponent. Damian et al. [Dam+24] formalized this as the *generative exponent*:

Definition 1.3 (Generative exponent [Dam+24])**.**

$$\ell^\star = \min\left\{j \geq 1 : \mathbb{E}_{z\sim\mathcal{N}(0,1)}\left[g(h^\star(z))\,\text{He}_j(z)\right] \neq 0 \text{ for some } g : \mathbb{R} \to \mathbb{R}\right\}, \tag{1.44}$$

where the minimization now ranges over all label transformations g as well as over the Hermite degree.

Since the choice $g = \mathrm{id}$ recovers Equation (1.31), one always has $\ell^\star \le \ell$; a nonlinear g can only lower the exponent, and frequently does so strictly. Damian et al. [Dam+24] showed that, when the *partial-trace* algorithm is run on a transformed label $g(y)$, the sample complexity of recovering the hidden direction is governed by $\ell^\star$ rather than ℓ: weak recovery is achieved in $d^{\ell^\star/2}$ samples, saturating the *Statistical Query* (SQ) lower bound. The algorithmic counterpart of this easy class was characterized concurrently by Dandi et al. [Dan+24b]: reusing each batch of an extensive ($\mathcal{O}(d)$) dataset even twice lets a two-layer network weakly recover any non-even single-index target (exactly the generative-exponent-one $\ell^\star = 1$ family) in $\mathcal{O}(d)$ samples, whereas single-pass SGD requires polynomially many more steps. The separation between ℓ and $\ell^\star$ can be substantial, as the following examples show.

The transformation g is target-dependent, but can be estimated from the data with a few guesses, as explained in the original work.

Example 1.6 (He_3)**.** He_3 *has information exponent* $\ell = 3$*, yet because it is odd one can take* $g(y) = \mathrm{sign}(y)$ *and correlate the transformed label with* He_1*, giving* $\ell^\star = 1$*. From a practical standpoint, this means that even the optimal partial-trace algorithm recovers* $\boldsymbol{w}$ *in* $\mathcal{O}(d^{3/2})$ *when restricted to linear queries of the label, but if it is allowed to first transform the label with* $g(y) = \mathrm{sign}(y)$ *it can recover* $\boldsymbol{w}$ *in* $\mathcal{O}(d)$*. See Figure 1.8(a) for a numerical illustration.*

Example 1.7 (He_4)**.** He_4 *has information exponent* $\ell = 4$*, but taking* $g(y) = y^2$ *allows one to correlate the transformed label with* He_2*, giving* $\ell^\star = 2$*. The complexity drops from* $\mathcal{O}(d^2)$ *under CSQ to* $\mathcal{O}(d \log d)$*; see Figure 1.8(b) for a numerical illustration.*

The generative exponent is *composition-invariant*: no further transformation of the label can lower it; $d^{\ell^\star/2}$ is a genuine barrier and not an artifact of the query class: problems with $\ell^\star > 2$ are hard even for the best first-order method, approximate message passing [Tro+25a]. That analysis in fact delivers more than the exponent: being optimal among first-order methods, approximate message passing fixes the *sharp* weak-recovery thresholds, i.e. the exact critical sample ratio n/d above which a target with $\ell^\star \le 2$ becomes weakly learnable, and shows that no finite ratio suffices once $\ell^\star > 2$.

Displaced gradient SGD achieves the generative exponent rate The generative exponent sets the complexity lower bound, but can a gradient-based algorithm achieve it *agnostically*, without prior knowledge of the optimal transformation g? The key mechanism is remarkably simple: *data reuse*. When the gradient step uses the same sample twice, as in the displaced gradient algorithm of Chapter 5, the update direction becomes a nonlinear function of y. Concretely, the displaced gradient first computes a perturbed weight $\tilde{\boldsymbol{w}} = \boldsymbol{w} - \rho \nabla_{\boldsymbol{w}} \ell$ and then evaluates the gradient at $\tilde{\boldsymbol{w}}$ on the same sample:

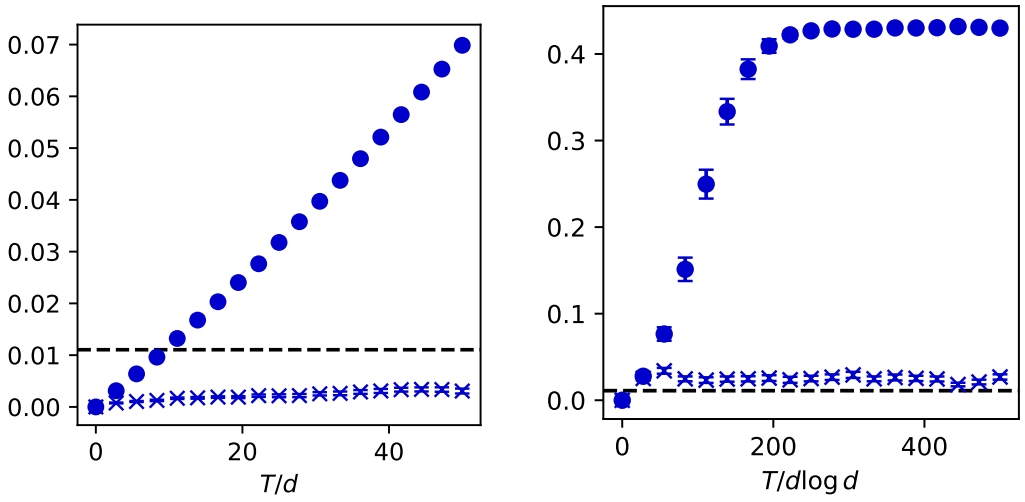
$$\nabla_{\boldsymbol{w}} \ell\big(f(\boldsymbol{x}; \tilde{\boldsymbol{w}}),\, y\big) = -\, y\, \sigma'(\tilde{\lambda})\, \frac{\boldsymbol{x}}{\sqrt{d}}, \qquad \tilde{\lambda} = \frac{\langle \tilde{\boldsymbol{w}}, \boldsymbol{x} \rangle}{\sqrt{d}}.$$

If ρ is positive the method is known as *extragradient*, and if ρ is negative it is known as *sharpness-aware minimization*.

With $\rho = \Theta(1)$, the perturbed pre-activation $\tilde{\lambda} = \lambda + \rho\, y\, \sigma'(\lambda)\, \|\boldsymbol{x}\|^2/d$ receives an $\mathcal{O}(1)$ perturbation that is a *nonlinear* function of y (because $\|\boldsymbol{x}\|^2/d \to 1$). Taylor-expanding σ' around λ generates higher-order correlational queries $\mathbb{E}[p(y)\, \varphi(\boldsymbol{x})]$ for a polynomial p determined by the activation

Chapter 5 states the same perturbation with $\rho = \rho_0/d$; that chapter works with the raw pre-activation $\langle \boldsymbol{w}, \boldsymbol{x} \rangle$ rather than the $1/\sqrt{d}$-normalized λ used here, and the two conventions agree once the extra factor of $\sqrt{d}$ from $\lambda = \langle \boldsymbol{w}, \boldsymbol{x} \rangle / \sqrt{d}$ is accounted for.

$$\nabla_{\boldsymbol{w}} \ell\big(f(\boldsymbol{x}; \tilde{\boldsymbol{w}}),\, y\big) \approx -\, \frac{\boldsymbol{x}}{\sqrt{d}} \left(y\, \sigma'(\lambda) + \rho\, y^2\, \sigma'(\lambda) \sigma''(\lambda) + \frac{\rho^2}{2}\, y^3\, \sigma'(\lambda)^2 \sigma'''(\lambda) + \cdots \right),$$



(a) $\ell = 3$, $\ell^\star = 1$: target He_3 (b) $\ell = 4$, $\ell^\star = 2$: target He_4

Figure 1.8: **Information exponent vs. generative exponent: separation in sample complexity** (from Chapter 5). Both panels plot the teacher–student overlap m as a function of the number of gradient steps T, for one-pass SGD (crosses) and the displaced gradient algorithm (circles) at several values of d. *(a)* Target He_3 has $\ell = 3$ but $\ell^\star = 1$ (the label can be sign-transformed to produce a linear-exponent query): one-pass SGD requires $\mathcal{O}(d^2)$ steps, while the displaced gradient recovers the teacher in $\mathcal{O}(d)$. *(b)* Target He_4 has $\ell = 4$ but $\ell^\star = 2$ (it is even, so $\ell^\star \geq 2$): one-pass SGD needs $\mathcal{O}(d^3)$ steps, while the displaced gradient achieves recovery in $\mathcal{O}(d \log d)$.

so the network implicitly synthesizes the label transformation, provided σ has higher-order derivatives. This result is formalized in the following theorem:

Theorem 1.3 (Informal version of Chapter 5)**.** *The displaced gradient algorithm of Chapter 5, which sees each sample exactly twice, achieves weak recovery of any polynomial single-index target with generative exponent $\ell^\star$ in*

$$n = \begin{cases} \mathcal{O}(d) & \ell^\star = 1, \\ \mathcal{O}(d \log d) & \ell^\star = 2. \end{cases} \tag{1.45}$$

The theorem is stated only for $\ell^\star \leq 2$ because if $h^\star$ is a polynomial, then $\ell^\star \leq 2$, as proved in Chapter 5. There is no guarantee of the exact same sample complexity for $\ell^\star > 2$, but it cannot be better than the SQ lower bound $d^{\ell^\star/2} \geq d^{\frac{3}{2}}$ (the last inequality holding because $\ell^\star \geq 3$ in this regime).

This is the same sample-complexity table as (1.33), with ℓ replaced by $\ell^\star$. Since $\ell^\star \leq \ell$, the displaced gradient is always at least as efficient as one-pass SGD and strictly better whenever $\ell^\star < \ell$.

In the multi-index setting, data reuse extends the staircase hierarchy. One-pass SGD follows the *CSQ staircase* of Abbe, Adserà, and Misiakiewicz [AAM23]: each phase aligns with the direction of smallest *directional information exponent* and the cost of each step scales as $d^{\ell_{\boldsymbol{v}}-1}$. With the displaced gradient, the staircase is governed by the generative exponent instead, enabling an *SQ staircase* over a strictly larger class of functions.

The same object is called the *grand staircase* by Troiani et al. [Tro+25a], in the setting of approximate message passing; we keep *SQ staircase* to stress that the climbable hierarchy is the one set by the generative (SQ) exponent.

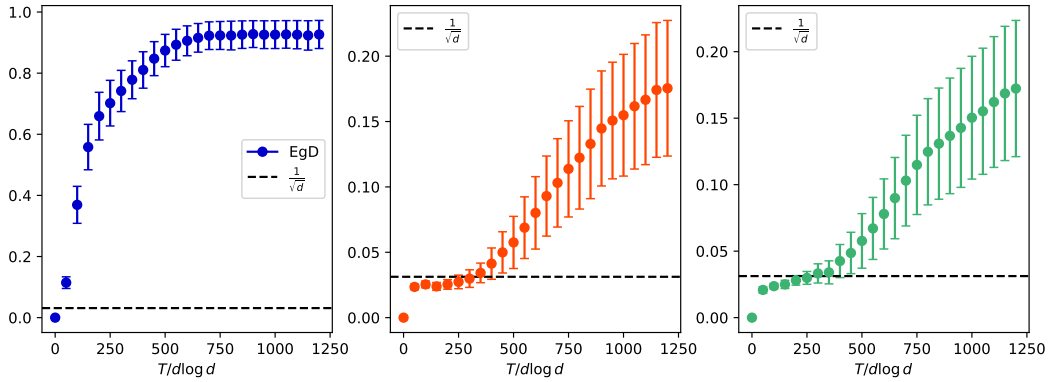


Figure 1.9: **An SQ staircase that SGD cannot climb** (from Chapter 5). Maximum cosine similarity between the student neurons and each teacher direction—$\boldsymbol{w}_1^\star$ (blue), $\boldsymbol{w}_2^\star$ (orange), $\boldsymbol{w}_3^\star$ (green)—as a function of the normalized iteration time, for the displaced gradient on the target $h^\star(\boldsymbol{\lambda}^\star) = \mathrm{He}_4(\lambda_1^\star) + \mathrm{sign}(\lambda_1^\star\lambda_2^\star\lambda_3^\star)$. Using repeated data gradient descent (EgD), the first direction is weakly recovered in $\mathcal{O}(d\log d)$ steps through its generative exponent $\ell^\star = 2$; that overlap then unlocks the remaining two in a further $\mathcal{O}(d\log d)$. One-pass SGD cannot start the climb at all: the two components have information exponents $\ell = 4$ and $\ell = 3$, so no rung is CSQ-accessible.

Example 1.8 (SQ staircase)**.** *A canonical example is*

$$h^\star(\boldsymbol{\lambda}^\star) = \mathrm{He}_4(\lambda_1^\star) + \mathrm{sign}(\lambda_1^\star\lambda_2^\star\lambda_3^\star).$$

The He_4 *component has* $\ell = 4$, $\ell^\star = 2$, *and the displaced gradient weakly recovers the first direction in* $\mathcal{O}(d\log d)$ *steps; once* $m_1 = \mathcal{O}(1)$, *the* sign *component, which has* $\ell = \ell^\star = 3$, *becomes learnable and is recovered in a second* $\mathcal{O}(d\log d)$ *phase. The same function is invisible to the CSQ staircase because its first direction has* $\ell_{\boldsymbol{v}} = 4$. *A numerical illustration is shown in Figure 1.9.*

Together, the three exponents trace the arc of this thesis: the information exponent fixes the cost of one-pass SGD on a single-index target, the leap exponent lifts it to the multi-index staircase, and the generative exponent marks the lower limit that becomes reachable once we step beyond a single pass.

Chapter 2

From high-dimensional & mean-field dynamics to dimensionless ODEs: **A unifying approach to SGD in two-layers networks**

Luca Arnaboldi, Ludovic Stephan, Florent Krzakala, and Bruno Loureiro

Abstract. This manuscript investigates the one-pass stochastic gradient descent (SGD) dynamics of a two-layer neural network trained on Gaussian data and labels generated by a similar, though not necessarily identical, target function. We rigorously analyse the limiting dynamics via a deterministic and low-dimensional description in terms of the sufficient statistics for the population risk. Our unifying analysis bridges different regimes of interest, such as the classical gradient-flow regime of vanishing learning rate, the high-dimensional regime of large input dimension, and the overparameterised "mean-field" regime of large network width, covering as well the intermediate regimes where the limiting dynamics is determined by the interplay between these behaviours. In particular, in the high-dimensional limit, the infinite-width dynamics is found to remain close to a low-dimensional subspace spanned by the target principal directions. Our results therefore provide a unifying picture of the limiting SGD dynamics with synthetic data.

2.1 Introduction

A detailed understanding of the performance of stochastic gradient descent (SGD) in neural networks is a major endeavor in machine learning, and significant progress was achieved in the context of large two-layer neural networks. In particular the optimization over wide two-layer neural networks can be rigorously studied using a well-defined partial differential equation (PDE) [MMN18; CB18; RV22; SS20]. A consequence of these results is the global convergence of overparametrized two-layer networks towards perfect learning provided that the number of hidden

neurons is large, the learning rate is sufficiently small, and enough data is at disposal. This line of work is commonly referred to as the *mean-field limit* of neural networks. The phenomenology in this regime was also studied for synthetic data with simple target functions by [MMM19; AAM22].

Interestingly, the SGD dynamics of two-layer neural networks trained on synthetic Gaussian data was considered as early as in the seminal work of [SS95b; SS95a; SS96], and has witnessed a renewal of activity over the last few years [Ver18; Gol+19; Vei+22; BGJ22]. However, differently from the mean-field limit, these works investigated the opposite limit of *fixed* hidden layer width and *diverging* data dimension, and studied the limiting SGD dynamics through a set of ordinary differential equations (ODEs).

Given these different limits, one may naturally wonder what is the relation, if any, between these sets of works. More generally, given data in dimension d and a two-layer network with p hidden units trained by SGD with a learning rate γ, one might inquire about the different regimes beside the mean-field ($p \to \infty$) and high-dimensional ($d \to \infty$) ones. This is the question investigated in this work. We consider a two-layer network trained on Gaussian data and labels given by a similar, though not necessarily identical, two-layer neural network target (hereafter also referred to as the *teacher*), and investigate the one-pass stochastic gradient descent (SGD) dynamics as a function of the relevant parameters d, p and γ. As summarized in Figure 2.1, we show that as long as $\gamma/dp \to 0^+$, a unifying deterministic description can be provided. In particular, our description recovers all the previously studied limits (mean field, high-dimensional and the classical gradient flow regime) and builds a bridge between them in a unified formalism. Namely, our **main contributions** are:

- Starting from the general approach of [SS95b; SS96], we build on the non-asymptotic results [Vei+22] and show how it allows to describe the entire phase space in Figure 2.1, that interpolates between the high-dimensional, classical, and mean-field limits.
- We unveil a remarkable dimension independence in the classical gradient-flow limit: once the initial conditions are given, the dynamics in terms of the sufficient statistics turns out to be *entirely* independent of the data dimension d.
- We explicitly construct the mean-field solution starting from the ODEs in the mean-field regime, bridging the hitherto different worlds of [SS95b; SS96] with the mean-field "hydrodynamic" approach. More precisely, for $p \to \infty$, we show how the ODEs simplify and give rise to a mean-field PDE, thanks to a decoupling of the learning dynamics that is found to remain close to a low-dimensional subspace spanned by the target principal directions.
- We further discuss how the SGD dynamics in the mean-field regime behave differently whether the input dimension d is small or large, leading to different finite-d and high-d dimensionless PDEs. This generalizes the recent "dimension-free" results in [HC23; AAM22].
- We provide a numerical solver for these equations. A GitHub repository with the code employed in the present work is available on GitHub repository[1]. Additionally, we discuss

[1] `https://github.com/IdePHICS/DimensionlessDynamicsSGD`

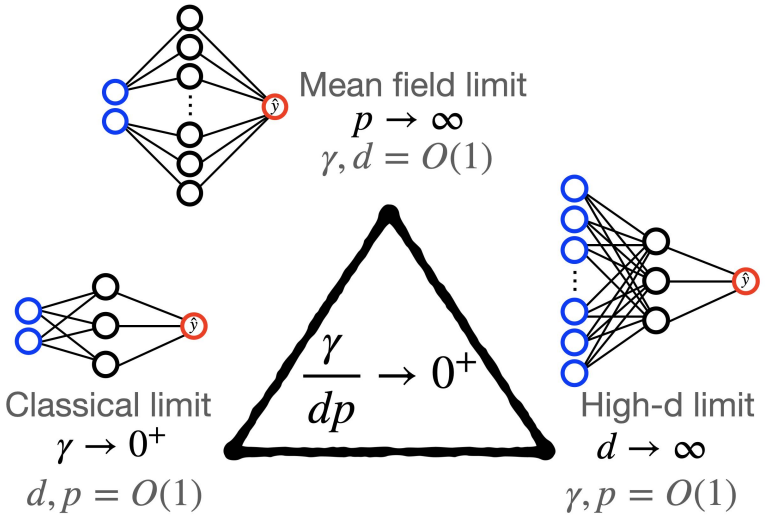


Figure 2.1: We discuss a unifying low-dimensional description of the one-pass SGD dynamics Equation (2.3) for two-layers neural networks as $\gamma/dp \to 0^+$. This includes, in particular, the mean-field ($p \to \infty$), the high-dimensional ($d \to \infty$) and the classical gradient flow ($\gamma \to 0$) limits.

the interesting case of quadratic activation that allows to drastically reduce the complexity of the ODEs & PDEs.

Related work — Stochastic gradient descent was first introduced [RM51] as a stochastic approximation method, and later applied as an approximation to population risk minimization in [BC03; BB07]. Its properties have been extensively studied for finite learning rate and input dimension in the strongly convex setting [Pfl86; MB11; DFB17; DDB20], and more recently in convex problems in the interpolating regime [MBB18; VBS19; Zou+21; VPF21], to cite a few. High-dimensional limits of SGD were studied in [Ver18; BGJ21] for non-convex, single-index models. Recently, [BGJ22] has generalized and abstracted this discussion.

In the context of two-layer neural networks, the high-dimensional limit of SGD draws back from the seminal work of [SS95b; SS95a; SS96] and was subsequently studied by many authors under different settings [BS95; CC95; BRW96; Gol+19; Gol+20; Gol+22; Ref+21]. The infinite-width (a.k.a. mean-field) limit of the SGD dynamics of two-layer neural networks was studied by [MMN18; CB18; RV22; SS20], who proved global convergence under certain conditions on the architecture and initialization. A bridge between these two limits was discussed by [Vei+22], who studied the joint limit where the hidden-layer width and the learning rate scale with the diverging input dimension.

Closer to us, dimension-free limits of the mean-field equations have been derived by [AAM22] for low-dimensional target functions in the hypercube and by [HC23] for ReLU networks when the target is invariant under certain symmetries. [BPF22] has proven global convergence of the gradient flow dynamics at finite width for orthogonal input data.

2.2 Setting

In this manuscript we consider a supervised learning regression task where we are given n independent samples $(\boldsymbol{x}^\nu, y^\nu)_{\nu\in[n]} \in \mathbb{R}^{d+1}$ from a probability distribution ρ. We are interested in the problem of learning the training data with a (fully-connected) two-layer neural network:

$$f_\Theta(\boldsymbol{x}) = \frac{1}{p}\sum_{i=1}^{p} a_i \sigma(\boldsymbol{w}_i^\top \boldsymbol{x}), \tag{2.1}$$

where $\Theta = (\boldsymbol{a}, \mathrm{W}) \in \mathbb{R}^{p(d+1)}$ denote the trainable parameters and $\sigma : \mathbb{R} \to \mathbb{R}$ the activation function. Since one of our goals is to connect with this line of work, for convenience we have adopted the mean-field normalization [MMN18; CB18; RV22; SS20]. As usual, training is performed via *empirical risk minimization*, where the statistician chooses a *loss function* $\ell : (f_\Theta(\boldsymbol{x}), y) \in \mathbb{R}^2 \mapsto \ell(f_\Theta(\boldsymbol{x}), y) \in \mathbb{R}_+$ penalizing deviations from the true labels and optimizes the training parameters Θ by minimizing the loss over the training data. As it is common in regression, we use the square loss $\ell(\hat{y}, y) \coloneqq {}^1\!/_2(\hat{y} - y)^2$, and focus on the *generalization error* or *population risk*:

$$\mathcal{R}(\Theta) \coloneqq \mathbb{E}_{(\boldsymbol{x},y)\sim\rho}\left[\frac{1}{2}(f_\Theta(\boldsymbol{x}) - y)^2\right] \tag{2.2}$$

Training algorithm: This empirical risk minimization problem being non-convex, different optimization algorithms might reach different minima. Our goal is to characterize the training dynamics of two-layer networks under *one-pass stochastic gradient descent*:

$$\Theta^{\nu+1} = \Theta^\nu - \gamma \nabla_\Theta \ell(f_{\Theta^\nu}(\boldsymbol{x}^\nu), y^\nu), \qquad \nu \le n \tag{2.3}$$

Note that in one-pass SGD, a fresh sample of data is used to estimate the gradient at each step, and therefore the quantity of data seen by the algorithm coincides with the number of steps. In particular, this means that at each step ν we have a random unbiased estimation of the population gradient which is uncorrelated to the previous step, defining a Markov chain. It is useful to rewrite Equation (2.3) by making explicit the effective noise of the process:

$$\Theta^{\nu+1} = \Theta^\nu - \gamma\nabla_\Theta \mathcal{R}(\Theta^\nu) + \gamma\varepsilon_\nu, \qquad \varepsilon_\nu \coloneqq \nabla_\Theta\left[\mathcal{R}(\Theta^\nu) - \ell(f_{\Theta^\nu}(\boldsymbol{x}^\nu), y^\nu)\right] \tag{2.4}$$

which is a zero mean random variable. Therefore, characterizing the training dynamics of one-pass SGD translates into characterizing this stochastic process.

Data model: Stochasticity in Equation (2.3) is induced by the draw of samples $(\boldsymbol{x}, y) \sim \rho$. In the following, we will assume that the input $\boldsymbol{x}^\nu$ are Gaussian $\boldsymbol{x}^\nu \sim \mathcal{N}\left(\mathbf{0}_d, {}^1\!/_d \mathrm{I}_d\right)$ while y^ν is drawn from the following generative model:

$$y^\nu = \frac{1}{k}\sum_{r=1}^{k} a_r^\star \sigma^\star(\boldsymbol{w}_r^{\star\top}\boldsymbol{x}^\nu) + \sqrt{\Delta} z^\nu, \qquad z^\nu \sim \mathcal{N}(0,1) \tag{2.5}$$

In other words, the target function $f_{\Theta^\star}$ is itself a two-layer neural network with parameters $\Theta^\star = (\boldsymbol{a}^\star, \mathrm{W}^\star) \in \mathbb{R}^{k(d+1)}$ and activation $\sigma^\star$. This setting, commonly refereed to as the *teacher-student* scenario, provides a rich data model for studying generalization, and has been employed both in the analysis of one-pass SGD [SS96; Gol+19; Vei+22] but also more broadly in high-dimensional statistics [Lou+21]. In particular, we will be mostly interested in the realizable scenario where $k \leq p$, and therefore the minimum of the population risk in Equation (2.2) is achieved by perfectly learning the target.

Technical assumptions: We assume that the SGD dynamics remains in a bounded subset of $\mathbb{R}^{p\times d}$:

Assumption 2.1. *On an event with high probability, the SGD iterates are bounded in the following sense: for some $K > 0$, we have*

$$\forall i \in [p], \quad \|\boldsymbol{w_i}\|^2 \leq K. \tag{2.6}$$

This assumption can be easily checked on either the simulations or their deterministic approximations (see below), or otherwise enforced with a weight decay (as in [WJ22]).

Assumption 2.2. *The student activation function σ is twice differentiable, with $\|\sigma^{(i)}\|_\infty \leq K$ for $i = 0, 1, 2$. The teacher activation $\sigma^\star$ is also upper bounded by K.*

Note that in some plots, we will sometimes use the function $\sigma(x) = x^2$, which does not satisfy this assumption. However, Assumption 2.1 ensures that we stay in a bounded subset of $\mathbb{R}^d$, hence we can replace σ by $\sigma \wedge K$ for K sufficiently large.

Simplifying assumptions: Since most of the interesting phenomenology happens at the hidden-layer, to lighten the discussion in the following we will focus on the case in which $a_r^\star = 1$ and $a_i^\nu = 1$ are fixed throughout learning and p is divisible by k. All of the discussion that follows can be readily generalized to the case in which $\boldsymbol{a}^\nu$ is learned and $p \geq k$ generically. We shall also assume that the teacher matrix $\mathrm{W}^\star$ is full rank; this can be avoided with a more careful definition of projections, but complicates the analysis.

2.3 The three limit regimes and their dimensionless description

As discussed before, the optimization problem introduced in Section 2.2 defines a non-convex optimization problem. Moreover, modern neural networks operate in a regime where both the data dimension d and the number of parameters in the network p are large. Therefore, characterizing the evolution of the weights $\Theta^\nu \in \mathbb{R}^{p(d+1)}$ amounts to studying $p(d+1)$ non-linear, coupled, non-convex stochastic process - a challenging problem even for numerical methods. As motivated in the introduction Section 2.1, in this section we derive a *tractable*, *low-dimensional* description for SGD in different regimes of practical interest.

2.3.1 Main concepts

Sufficient statistics: A first observation is that the performance of the predictor $(\Theta^\nu)_{\nu \le n}$ at iteration $\nu \le n$ only depends on the statistics of the student and teacher pre-activations

$$\boldsymbol{\lambda}^\nu \coloneqq \mathrm{W}^\nu \boldsymbol{x}^\nu \in \mathbb{R}^p, \quad \boldsymbol{\lambda}^{\star\nu} \coloneqq \mathrm{W}^\star \boldsymbol{x}^\nu \in \mathbb{R}^k \tag{2.7}$$

Moreover, since $\boldsymbol{x}^\nu$ is Gaussian and independent of $(\mathrm{W}^\nu, \mathrm{W}^\star)$, the pre-activations are jointly Gaussian vectors $(\boldsymbol{\lambda}^\nu, \boldsymbol{\lambda}^{\star\nu}) \sim \mathcal{N}(\mathbf{0}_{p+k}, \Omega^\nu)$ with covariance:

$$\Omega^\nu \coloneqq \begin{pmatrix} \mathrm{Q}^\nu & \mathrm{M}^\nu \\ \mathrm{M}^{\nu\top} & \mathrm{P} \end{pmatrix} = \begin{pmatrix} 1/d \mathrm{W}^\nu \mathrm{W}^{\nu\top} & 1/d \mathrm{W}^\nu \mathrm{W}^{\star\top} \\ 1/d \mathrm{W}^\star \mathrm{W}^{\nu\top} & 1/d \mathrm{W}^\star \mathrm{W}^{\star\top} \end{pmatrix} \in \mathbb{R}^{(p+k)\times(p+k)} \tag{2.8}$$

which is the *sufficient statistics* matrix for the population risk in Equation (2.2). Massaging Equation (2.3), we can derive a closed set of stochastic processes governing the evolution of the sufficient statistics:

$$\begin{aligned} M_{ir}^{\nu+1} - M_{ir}^\nu &= \frac{\gamma}{pd} \sigma'(\lambda_i^\nu) \lambda_r^{\star\nu} \mathcal{E}^\nu \\ Q_{ij}^{\nu+1} - Q_{ij}^\nu &= \frac{\gamma}{pd} \left(\sigma'(\lambda_i^\nu)\lambda_j^\nu + \sigma'(\lambda_j^\nu)\lambda_i^\nu \right) \mathcal{E}^\nu + \frac{\gamma^2 \, ||\boldsymbol{x}^\nu||_2^2}{p^2 d} \sigma'(\lambda_i^\nu)\sigma'(\lambda_j^\nu) \mathcal{E}^{\nu 2} \end{aligned} \tag{2.9}$$

where we defined for convenience the displacement vector

$$\mathcal{E}^\nu \coloneqq \frac{1}{k} \sum_{r=1}^k \sigma^\star(\lambda_r^{\star\nu}) - \frac{1}{p} \sum_{j=1}^p \sigma(\lambda_j^\nu) + \sqrt{\Delta} z^\nu. \tag{2.10}$$

Note that so far we have made no approximations: these equations are exact, and allow us to trade the $p(d+1)$ dimensional process for Θ^ν in Equation (2.3) for a $p(k+p)$ dimensional process for $(\mathrm{M}^\nu, \mathrm{Q}^\nu)$. This can be particularly convenient if $d \gg k, p$.

The process in Equation (2.9) has been previously studied in different limits and particular cases. For instance, [TV23] studied the stochastic dynamics in particular case of $k = p = 1$ and $\sigma(x) = x^2$, also known as *phase retrieval*, and [BGJ21] extended this discussion for arbitrary σ. More important to our work, [SS95b; SS96] has shown that this process admits a deterministic limit when $d \to \infty$ at fixed p, γ, characterized by the following ODE:

$$\frac{\mathrm{dM}}{dt} = \Psi^{(\mathrm{M})}(\Omega), \qquad \frac{\mathrm{dQ}}{\mathrm{d}t} = \Psi^{(\mathrm{GF})}(\Omega) + \frac{\gamma}{p} \Psi^{(\mathrm{Var})}(\Omega), \tag{2.11}$$

where the right-hand side functions are defined by the following equations.

$$\begin{aligned} \Psi_{ir}^{(\mathrm{M})}(\Omega) &= \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)} \left[\sigma'(\lambda_i) \lambda_r^\star \mathcal{E} \right] \\ \Psi_{ij}^{(\mathrm{GF})}(\Omega) &= \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)} \left[\left(\sigma'(\lambda_i)\lambda_j + \sigma'(\lambda_j)\lambda_i \right) \mathcal{E} \right] \\ \Psi_{ij}^{(\mathrm{Var})}(\Omega) &= \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)} \left[\sigma'(\lambda_i)\sigma'(\lambda_j) \mathcal{E}^2 \right] \end{aligned} \tag{2.12}$$

As will be discussed later in Section 2.3.2, the superscript notation for the right-hand side is suggestive of their interpretation. This convergence was made rigorous by [Gol+19] and [Vei+22], who showed the following non-asymptotic result:

Theorem 2.1 ([Vei+22]). *We place ourselves under Assumptions 2.1-2.2. Let Ω^ν be the random process of Equation* (2.9)*, and $\Omega(t)$ the solution to the ODE* (2.11) *with starting point $\Omega(0) = \Omega^0$. Define the step-size $\delta t = \gamma/pd$, and assume that $\gamma/p = O(1)$. Then there exists a constant $C > 0$ such that for any $\nu \geq 0$,*

$$\|\Omega^\nu - \Omega(\nu\delta t)\|_\infty \leq e^{C\nu\delta t}\sqrt{\frac{\gamma}{pd}} \tag{2.13}$$

[Vei+22] also described the behavior of Ω for various choices of γ and p. In particular, when $\gamma/p \ll 1$, Equations (2.11) reduce to the following simpler ones:

$$\frac{\mathrm{dM}}{\mathrm{d}t} = \Psi^{(\mathrm{M})}(\Omega), \qquad \frac{\mathrm{dQ}}{\mathrm{d}t} = \Psi^{(\mathrm{GF})}(\Omega) \tag{2.14}$$

Our key observation is that the deterministic description above is valid beyond the high-dimensional limit $d \to \infty$ on which previous works [SS95b; Gol+19; Vei+22] have focused. Indeed, Theorem 2.1 is non-asymptotic in (d, p, γ), and can thus be applied in any setting where $\gamma/pd \ll 1$. This is leveraged to provide a tractable, low-dimensional description of SGD in different scenarios of interest which we summarize in Figure 2.1.

2.3.2 The classical regime

The first and most well-studied scenario is the classical regime in which $\gamma \to 0^+$ at fixed dimensions $d, p = O(1)$. Defining the continuous weight $\Theta^\nu = \Theta(\nu\delta t)$ via linear interpolation, a classical result from stochastic optimization [RM51] is that one-pass SGD converges to gradient flow on the population risk:

$$\frac{\mathrm{d}\Theta(t)}{\mathrm{d}t} = -\nabla_\Theta \mathcal{R}(\Theta(t)) \tag{2.15}$$

Or in words: the effective SGD noise ε_ν in Equation (2.4) is subleading in this limit. Note that this is a deterministic ordinary differential equation of dimension $p(d+1)$. Since $d, p = O(1)$, if they are small Equation (2.15) provides a computationally efficient description of the SGD dynamics, since it can be easily implemented and solved in a computer. Nonetheless, an alternative description can be derived from the sufficient statistics of Equation (2.8). Indeed, Theorem 2.1 guarantees that in the limit $\gamma \to 0^+$, the stochastic process in Equation (2.9) converges to the deterministic limit of Equation (2.14). This ODE can be easily seen to be equivalent to the one of (2.15), through the following identity:

$$\frac{\mathrm{d}\langle \boldsymbol{w_i}, \boldsymbol{w_j}\rangle}{\mathrm{d}t} = \left\langle \boldsymbol{w_i}, \frac{\mathrm{d}\boldsymbol{w_j}}{\mathrm{d}t}\right\rangle + \left\langle \frac{\mathrm{d}\boldsymbol{w_i}}{\mathrm{d}t}, \boldsymbol{w_j}\right\rangle$$

This gives rise to ordinary differential equations in $p(k+p)$ parameters. Therefore, depending on the values of (d, p), it can offer a more compact description of the evolution of the performance of the predictor than Equation (2.15).

As it was previously hinted by the notation, Equation (2.14) also provides an intuitive interpretation of the right-hand side of the stochastic process (2.9). Indeed, the terms proportional to γ/pd

in Equation (2.9) correspond exactly to the terms inside the expectation in Equation (2.14), and correspond to the projection of the population gradient along the weights $(\Theta, \Theta_\star)$. The remaining term, which is proportional to $\gamma^2/p^2 d$, comes from the variance of the effective noise ε, which is subleading in the limit $\gamma \to 0^+$. This agrees with the characterization of the terms given in [BGJ22], where an additional Brownian motion correction term at finer scales is also derived.

In Figure 2.2 (left) we plot the trajectories of individual neurons in the space spanned by the two target neurons ($k = 2$); $\mathrm{M}_{jr}/\sqrt{\mathrm{Q}_{jj}\mathrm{P}_{rr}}$ is the (normalized) scalar product between $\boldsymbol{w}_j$ and $\boldsymbol{w}_r^*$. While, initially, all neurons are moving towards the superposition of the two teacher weights $((\sqrt{2}/2, \sqrt{2}/2)$, yellow dot), in the last phase of learning they "specialize" and split evenly to one of the two ($(1, 0), (0, 1)$, red squares). Note how the ODE trajectories closely follow the simulated ones, given the specific initialization.

Finally, our results imply a remarkable **dimension independence** property: Differently from (2.15), the alternative description (2.14) turns out to be independent of the data dimension d. Given the initial conditions of the sufficient statistics (the overlaps), then the trajectories will behave **exactly in the same way** whether d is large or small (see the illustration in Figure 2.2 (right)). This remarkable property is a direct consequence of the Gaussianity assumption on the data. Note, of course, that dimensionality still plays a crucial role through the initialization. Indeed, for the typically employed random initialization $\theta_i^0 \sim \mathcal{N}(0_d, \varsigma^2 \mathrm{I}_d)$, the initial correlation between the hidden-units and the target, parameterized by $M_{rj}^0 \sim 1/\sqrt{d}$, explicitly depends on d. Since $M = 0$ is often a fixed-point of the dynamics, $d = O(1)$ or $d \gg 1$ will lead effectively to different behaviors.

2.3.3 The high-dimensional regime

Modern machine learning practice often involves high-dimensional data. As early as in [SS95b; SS96] it has motivated the study of the $d \to \infty$ limit of the SGD dynamics (2.4), under the assumption of fixed learning rate and model complexity $p, \gamma = O(1)$. This setting has witnessed a renewal of interest recently [Ver18; Gol+19; Vei+22; BGJ22]. In particular, a remarkable phenomenon arises: differently from the classical limit Equation (2.14) discussed above, in high-dimension the variance term induced by the SGD effective noise yields an explicit contribution to the limiting dynamics (2.11). This term can yield a finite risk contribution at large times even for architectures for which the population gradient flow would otherwise converge to zero population risk (i.e. perfect learning of the target $f_{\Theta^\star}$), so that the large time dynamics plateau at finite risk (see for instance [Gol+19; BGJ22]). This is a major difference between the classical and high-dimensional regime.

For strongly convex problems, it is known that SGD with fixed learning rate γ converges to a stationary distribution of variance $\propto \gamma$ [Pfl86; DDB20], leading to an asymptotic risk that closely resembles the one observed by [SS95b] in the high-dimensional regime. This suggests a similar phenomenology in the basin of the global minima, although making this statement precise is challenging due to the non-convexity of the risk for two-layer networks. As noted by [Vei+22], this noise term is subleading in γ/p, and can be mitigated by either taking $\gamma \to 0^+$ (i.e. seeing a lot of data) or overparametrizing $p \to \infty$. However, since Equations (2.11) are a system of $p(p + k)$ ordinary differential equations, they become intractable in the limit $p \to \infty$, which is a major

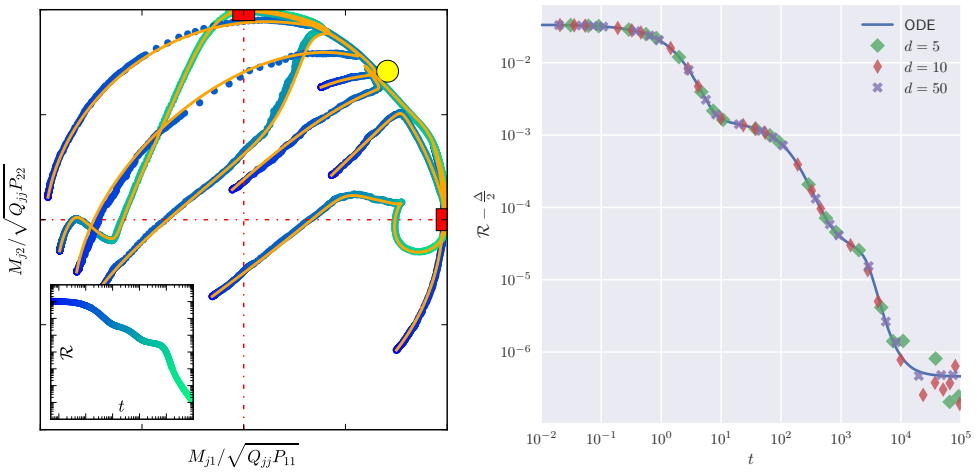


Figure 2.2: **Dimension independence of the dynamics in the classical regime**: Comparison between the simulated SGD dynamic and the analytical one obtained by integrating the differential equations. All the plots are using $\sigma = \mathrm{erf}(\cdot/\sqrt{2})$. (Left) Trajectories for the cosine similarity between each network neuron and the target; time evolution in simulation is blue to green, ODE trajectories in orange ($k = 2, p = 10, \gamma = 5 \times 10^{-2}, \Delta = 0$). (Right) Different low-dimensional simulations, starting from the same identical initial condition ($k = 5, p = 10, \gamma = 5 \times 10^{-2}, \Delta = 10^{-3}$).

shortcoming of this description.

2.3.4 The overparametrized regime

In both the classical and high-dimensional regimes, the effective description of the SGD dynamics rely on quantities which scale with the hidden-layer width p, and therefore they are not adequate to wide models. Yet, in many scenarios of interest we need to deal with wide, overparametrized networks. The problem of finding an effective low-dimensional description of one-pass SGD for the overparametrized regime $p \to \infty$ was first addressed by [MMN18; CB18; RV22; SS20]. The key idea in this line of work is to define an empirical density over the weights $\theta_i^\nu := (a_i^\nu, \boldsymbol{w}_i^\nu)$:

$$\hat{\mu}_p^\nu(\boldsymbol{\theta}) = \frac{1}{p}\sum_{i=1}^{p} \delta(\boldsymbol{\theta} - \boldsymbol{\theta}_i^\nu) \tag{2.16}$$

and to derive a closed-form update for the density $\hat{\mu}_p^\nu(\boldsymbol{\theta})$ from the SGD update of the weights, Equation (2.4). In the limit $p \to \infty$, those works have shown that the empirical density converges to an asymptotic density over $\mathbb{R}^{d+1}$, which for sufficiently small learning rate satisfies a partial differential equation (PDE) that became known in the literature as the *mean-field limit*. Drawing from the theory of PDEs and optimal transport, this description allowed for the derivation of important mathematical guarantees on the dynamics, such as the global convergence of SGD for two-layer neural networks.

However, the empirical measure $\hat{\mu}_p^\nu$ is defined on $\mathbb{R}^{d+1}$, so a problem still remains when d is large. Indeed, as remarked in [BC22] it remains challenging to draw quantitative results from this description except for considerably low-dimensional data. However, since in our setting the target function (2.5) only acts on a low-dimensional subspace of $\mathbb{R}^d$, it is possible to exploit the symmetries of the problem to derive an approximation of constant dimension. This low-dimensional equivalent stems from invariance properties of mean-field equations, which were also used in [AAM22] and studied in depth in [HC23]. However, [AAM22] only considers the $d \to \infty$ limit, while we derive a limit that is valid for any value of d. [HC23], on the other hand, is closer to our work (see e.g. their Lemma 4.2), but only handles the approximation of the dynamics by PDEs instead of ODEs.

Decomposing the dynamics: The starting point of this low-dimensional description is the decomposition of W as

$$W = W_{\text{proj}} + W^\perp, \tag{2.17}$$

where W_{proj} is the orthogonal projection on the teacher vectors $W^\star$. This projection can be expressed using the sufficient statistics defined in Equation (2.8):

$$W = MP^{-1}W^\star + W^\perp. \tag{2.18}$$

Similar to (2.7), we can then define the orthogonal pre-activations and its covariance matrix:

$$\boldsymbol{\lambda}^{\perp\nu} = W^{\perp\nu}\boldsymbol{x}^\nu \tag{2.19}$$

$$Q^\perp = {}^1\!/_d W^\perp (W^\perp)^\top = Q - MP^{-1}M^\top. \tag{2.20}$$

Since we are in a regime where $\gamma/p \ll 1$, the ODE approximation of SGD corresponds to Equation (2.14). From there, with a little algebra (see Appendix B.2), we can derive the corresponding equations for $Q^\perp$:

$$\frac{dQ_{ij}^\perp}{dt} = \mathbb{E}_{(\boldsymbol{\lambda}^\perp, \boldsymbol{\lambda})}\left[\left(\sigma'(\lambda_i)\lambda_j^\perp + \sigma'(\lambda_j)\lambda_i^\perp\right)\mathcal{E}\right] \coloneqq \Psi_{ij}^\perp(\Omega). \tag{2.21}$$

Low-dimensional approximation: Informally, the interesting part of the dynamics happens in a low-dimensional space: the one spanned by the target weights $W^\star$. The remainder of the dynamics only depends on the student-student vector interactions, which are orthogonally invariant. We therefore make the following assumption to enforce this invariance at the start:

Assumption 2.3. *The initial vectors $(\boldsymbol{w}_1, \dots, \boldsymbol{w}_p)$ are drawn i.i.d from an orthogonally invariant and ${}^{K^2}\!/_d$-subgaussian distribution μ_0, for some constant $K > 0$.*

We show in the appendix how we can then approximate the dynamics by only tracking the evolution of M and $\operatorname{diag}(Q^\perp)$, and using the following ansatz:

$$\boldsymbol{w}_i^\perp \approx \sqrt{d\, q_{ii}^\perp} \cdot \boldsymbol{g}_i, \tag{2.22}$$

where $\boldsymbol{g}_i$ are i.i.d uniform random variables on $\mathcal{S}^{d-k-1}$ independent from the $q_{ii}^{\perp}$. More precisely, we consider the reduced parameters $\tilde{\Theta} = (M, q) \in \mathbb{R}^{p(k+1)}$, and the following mean-field equivalent of the overlaps:

$$\tilde{\Omega} = \begin{pmatrix} \tilde{Q} & M \\ M^\top & P \end{pmatrix}, \quad \tilde{Q} = MP^{-1}M^\top + D_{\sqrt{q}}\Xi D_{\sqrt{q}} \tag{2.23}$$

where $D_{\sqrt{q}}$ is the diagonal matrix whose entries are the $\sqrt{q_i}$, and Ξ is a *random* matrix with independent entries such that

$$\Xi_{ii} = 1, \quad \Xi_{ij} = \langle \boldsymbol{g}, \boldsymbol{g'} \rangle \quad \text{with} \quad \boldsymbol{g}, \boldsymbol{g'} \sim \text{Unif}\left(\mathbb{S}^{d-k-1}\right)$$

Then, the mean-field ODEs read

$$\frac{\mathrm{d}M}{\mathrm{d}t} = \mathbb{E}_\Xi \left[\Psi^{(M)}(\tilde{\Omega})\right] \quad \frac{\mathrm{d}q_i}{\mathrm{d}t} = \mathbb{E}_\Xi \left[\Psi_{ii}^{\perp}(\tilde{\Omega})\right]. \tag{2.24}$$

Similarly, given the parameters Θ, the risk is computed as

$$\mathcal{R}(\tilde{\Theta}) = \mathbb{E}_\Xi \left[\mathcal{R}(\tilde{\Omega})\right] \tag{2.25}$$

The consistency of this approximation is given by the following theorem:

Theorem 2.2. *Under Assumptions 2.2 and 2.3, let* $\Omega(t)$ *and* $\tilde{\Theta}(t)$ *denote the solutions of the ODES* (2.14) *and* (2.24)*, respectively. Then with probability at least* $1 - e^{-z^2}$ *on the initialization:*

$$\sup_{t\in[0,T]} \left|\mathcal{R}((t)) - \mathcal{R}(\tilde{(t)})\right| \leq Ce^{CT}\left(\sqrt{\log(pT)} + z\right)/\sqrt{p}.$$

The proof is given in Appendix B.3. It uses key elements from the mean-field study of [MMM19]. Indeed, both the solutions of (2.14) & (2.24) can be viewed as the "particle dynamics" approximations (see e.g. [Che17]) of two mean-field PDEs $\mu_t, \tilde{\mu}_t$ on the space of network weights $\boldsymbol{w} \in \mathbb{R}^d$ and the space of reduced parameters $(\boldsymbol{m}, q) \in \mathbb{R}^{k+1}$, respectively. In turn, the invariance property of μ_0 extends to $\hat{\mu}_t$ for every $t \geq 0$, which implies that $\mathcal{R}(\mu_t) = \mathcal{R}(\tilde{\mu}_t)$.

Remark 2.1. *We do not imply in any way, shape or form that Equation* (2.22) *represents the actual distribution of the* $\boldsymbol{w}_i^{\perp}$*, or that (as in* (2.23)*) the entries of* $Q^{\perp}$ *are independent. Rather, it is the structure of the update equations that allows for this approximation to hold.*

The high-dimensional limit of mean-field: When $d \to \infty$, the off-diagonal entries of the matrix Ξ are of order $1/\sqrt{d}$, which suggests that they can be neglected safely. We therefore define the following equivalent of $\tilde{\Omega}$, which does not depend on auxiliary random variables:

$$\bar{\Omega} = \begin{pmatrix} \bar{Q} & M \\ M^\top & P \end{pmatrix}, \quad \bar{Q} = MP^{-1}M^\top + \text{diag}(q) \tag{2.26}$$

The following lemma then holds:

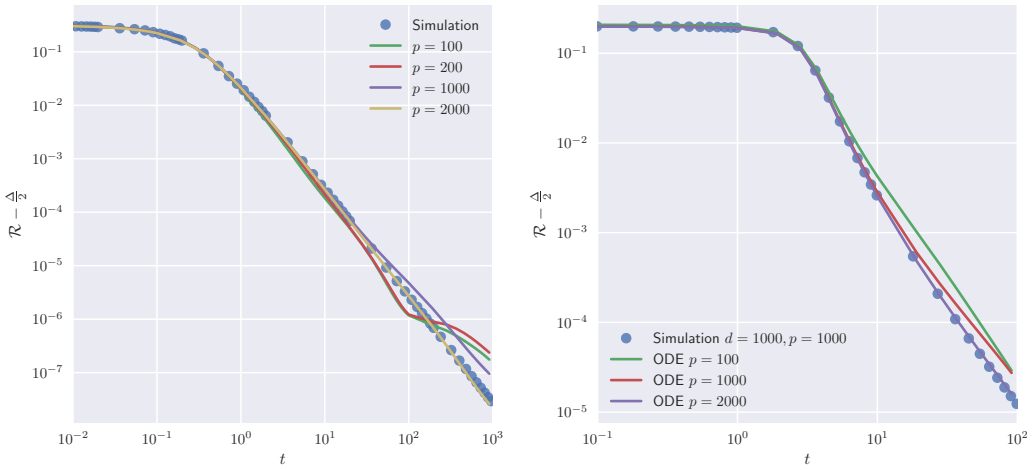


Figure 2.3: **The mean-field regime:** Comparison between population risks of the simulated learning dynamic and the corresponding deterministic evolution, for (left) mean-field low-dimensional regime with squared activation function ($k = 2, d = 5, \gamma = 1, \Delta = 0$) and (right) mean-field high-dimensional regime. Apart from finite size effects, there is agreement between ODEs and simulated dynamics ($k = 5, d = 1000, \gamma = 10$).

Lemma 2.1. *For any* $(M, q) \in \mathbb{R}^{p(k+1)}$*, we have:*

$$\left\| \mathbb{E}_\Xi \left[\Psi^{(M)}(\tilde{\Omega}) \right] - \Psi^{(M)}(\bar{\Omega}) \right\|_\infty \leq \frac{C}{\sqrt{d}}, \text{and} \left\| \mathbb{E}_\Xi \left[\Psi^{(\perp)}(\tilde{\Omega}) \right] - \Psi^{(\perp)}(\bar{\Omega}) \right\|_\infty \leq \frac{C}{\sqrt{d}}. \tag{2.27}$$

We thus define the high-dimensional equivalent of (2.24):

$$\frac{\mathrm{d}M}{\mathrm{d}t} = \Psi^{(M)}(\bar{\Omega}) \qquad \frac{\mathrm{d}q_i}{\mathrm{d}t} = \Psi^{\perp}_{ii}(\bar{\Omega}). \tag{2.28}$$

The same propagation of perturbations arguments discussed in Appendix B of [Vei+22], and the Lipschitz property of the risk, easily imply the following theorem:

Theorem 2.3. *Under Assumptions 2.2 & 2.3, let* $\Omega(t)$ *and* $\bar{\Theta}(t)$ *denote the solutions of the ODES* (2.14) *and* (2.28)*, respectively. Then with probability at least* $1 - e^{-z^2}$ *on the initialization:*

$$\sup_{t \in [0,T]} \left| \mathcal{R}(\Omega(t)) - \mathcal{R}(\bar{\Theta}(t)) \right| \leq C e^{CT} \left(\frac{\sqrt{\log(pT)} + z}{\sqrt{p}} + \frac{1}{\sqrt{d}} \right)$$

This approximation eliminates the need to compute expectations in (2.24). We now argue that this phenomenon is not only a consequence of rotation invariance, but also a simple concentration result. Indeed, irrespective of Assumption 2.3, we show the following result:

$$\left| \mathbb{E}_{\boldsymbol{\lambda}^\star, \boldsymbol{\lambda}^\perp \sim \mathcal{N}(\mathbf{0}, \Omega)} \left[\sigma(\lambda_i) \lambda_i^\perp \mathcal{E} \right] - \mathbb{E}_{\boldsymbol{\lambda}^\star, \boldsymbol{\lambda}^\perp \sim \mathcal{N}(\mathbf{0}, \bar{\Omega})} \left[\sigma(\lambda_i) \lambda_i^\perp \mathcal{E} \right] \right| \leq c \sqrt{Q^\perp_{ii} \cdot \frac{\|Q^\perp\|_{\mathrm{op}}}{p}}, \tag{2.29}$$

For a random sub-gaussian initialization, classical matrix concentration arguments (see [Ver18], Theorem 4.4.5) imply

$$\|Q^{\perp}\|_{\mathrm{op}} = O\left(1 + \frac{p}{d}\right)$$

and common sense arguments (the presence of an attracting force towards $Q^{\perp} = 0$) indicates that this quantity stays within the same order of magnitude. On the other hand, by Assumption 2.1, the Q_{ii} (and hence the $Q^{\perp}_{ii}$) remain bounded by an absolute constant during the trajectory. We therefore expect, using standard results from ODE perturbation, that

$$\|\bar{\Omega}(t) - \tilde{\Omega}(t)\|_{\infty} \leq e^{Ct}\left(\sqrt{\frac{1}{p}} + \sqrt{\frac{1}{d}}\right). \tag{2.30}$$

Contrary to the low-dimensional regime and the theorem above, this derivation does not require any rotation invariance of the $\boldsymbol{w}_i^{\perp}$; simply, the averaging properties of $\Psi^{(M)}$ are enough to show direct concentration properties on the dynamics.

It is instructive to look at a concrete example where the phenomenology discussed above becomes explicit. Perhaps the simplest one is given by the square activation $\sigma(x) = x^2$, for which the expectation in Equation (2.12) can be explicitly expressed in terms of polynomials of the covariance matrices $(\mathrm{Q}^{\nu}, \mathrm{M}^{\nu}, \mathrm{P})$ (see Appendix B.5 for the derivation):

$$\begin{aligned}
\Psi^{(\mathrm{M})}(\Omega) &= 2\left(\frac{\mathrm{Tr}\,[\mathrm{P}]}{k} - \frac{\mathrm{Tr}\,[\mathrm{Q}]}{p}\right)\mathrm{M} + 4\left(\frac{\mathrm{MP}}{k} - \frac{\mathrm{QM}}{p}\right) \\
\Psi^{(\mathrm{GF})}(\Omega) &= 4\left(\frac{\mathrm{Tr}\,[\mathrm{P}]}{k} - \frac{\mathrm{Tr}\,[\mathrm{Q}]}{p}\right)\mathrm{Q} + 8\left(\frac{\mathrm{MM}^{\top}}{k} - \frac{\mathrm{Q}^2}{p}\right) \\
\Psi^{(\perp)}(\Omega) &= 4\left(\frac{\mathrm{Tr}\,[\mathrm{P}]}{k} - \frac{\mathrm{Tr}\,[\mathrm{Q}]}{p}\right)\mathrm{Q}^{\perp} - \frac{4}{p}\left(\mathrm{QQ}^{\perp} + \mathrm{Q}^{\perp}\mathrm{Q}\right).
\end{aligned} \tag{2.31}$$

Similarly, the population risk as a function of the sufficient statistics reads:

$$\mathcal{R}(\Omega) = \frac{\mathrm{Tr}\,[\mathrm{P}]^2 + 2\,\mathrm{Tr}\left[\mathrm{P}^2\right]}{2k^2} - \frac{\mathrm{Tr}\,[\mathrm{P}]\,\mathrm{Tr}\,[\mathrm{Q}] + 2\,\mathrm{Tr}\left[\mathrm{MM}^{\top}\right]}{pk} + \frac{\mathrm{Tr}\,[\mathrm{Q}]^2 + 2\,\mathrm{Tr}\left[\mathrm{Q}^2\right]}{2p^2} + \frac{\Delta}{2}. \tag{2.32}$$

The high-dimensional mean field limit is then particularly simple. Recalling the definition of $\bar{\Omega}$ in Equation (2.26), we obtain the explicit equations by replacing Ω with $\bar{\Omega}$. The situation is, however, different for the low-dimensional mean-field limit, due to the randomness introduced by matrix Ξ. The ODEs and the risk, given the parameters $\tilde{\Theta}$, are

$$\begin{aligned}
\mathbb{E}_{\Xi}\left[\Psi^{(M)}(\tilde{\Omega})\right] &= \Psi^{(M)}(\bar{\Omega}) \qquad \mathbb{E}_{\Xi}\left[\Psi^{(\mathrm{noise})}(\tilde{\Omega})\right] = \Psi^{(\mathrm{noise})}(\bar{\Omega}) \\
\mathbb{E}_{\Xi}\left[\Psi^{\perp}_{ii}(\tilde{\Omega})\right] &= \Psi^{\perp}_{ii}(\bar{\Omega}) - \frac{8}{p}\frac{\sum_{j=1,j\neq i}^{p} q_j}{d-k} q_i \\
\mathbb{E}_{\Xi}\left[\mathcal{R}(\tilde{\Omega})\right] &= \mathcal{R}(\bar{\Omega}) + \frac{1}{p^2}\frac{\sum_{i=1}^{p}\sum_{j=1,j\neq i}^{p} q_i q_j}{d-k}.
\end{aligned} \tag{2.33}$$

In this case, the low-dimensional corrections are therefore just additive terms. As expected, these corrections vanish when $d \to \infty$, where we fall back to the high-dimensional mean-field. Conversely, when $d = k$ the correction diverges, but we don't need to track q anymore since the teacher weights are spanning the whole space $\mathbb{R}^d$, and the orthogonal space is null; hence $\mathrm{Q}^{\perp} = 0$ and this is a stable point of the dynamics. In Figure 2.3 we show some numerical experiments using this activation function; for a more detailed discussion see Appendix B.6.

2.4 Conclusion

Our work provides a comprehensive analysis of the one-pass SGD dynamics of two-layer neural networks. The study bridges different regimes of interest, offers a unifying picture of the limiting SGD dynamics, sheds light on the behavior of neural networks trained on synthetic data and, we believe, provides a useful tool for further investigations of the performance of these networks.

Acknowledgements

We thank Francis Bach, Gérard Ben-Arous, Lenaic Chizat, Theodor Misiakiewicz & Lenka Zdeborová for valuable discussions. We acknowledge funding from the Swiss National Science Foundation grant SNFS OperaGOST, 200021_200390 and the *Choose France - CNRS AI Rising Talents* program.

Chapter 3

Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD

Luca Arnaboldi, Florent Krzakala, Bruno Loureiro, and Ludovic Stephan

ABSTRACT. This study explores the sample complexity for two-layer neural networks to learn a generalized linear target function under Stochastic Gradient Descent (SGD), focusing on the challenging regime where many flat directions are present at initialization. It is well-established that in this scenario $n = O(d \log d)$ samples are typically needed. However, we provide precise results concerning the pre-factors in high-dimensional contexts and for varying widths. Notably, our findings suggest that overparameterization can only enhance convergence by a constant factor within this problem class. These insights are grounded in the reduction of SGD dynamics to a stochastic process in lower dimensions, where escaping mediocrity equates to calculating an exit time. Yet, we demonstrate that a deterministic approximation of this process adequately represents the escape time, implying that the role of stochasticity may be minimal in this scenario.

3.1 Introduction

In this manuscript we are interested in the supervised task of learning the following target function:

$$y = \sigma_\star\left(w_\star^\top x\right) + \sqrt{\Delta} z, \qquad x \sim \mathcal{N}(0, {}^1\!/_d I_d), \qquad z \sim \mathcal{N}(0,1). \tag{3.1}$$

This target belongs to a general class of models known as *single-index* or *generalized linear* models, where the labels depend on the covariates $x \in \mathbb{R}^d$ only through its projection on a fixed direction $w_\star \in \mathbb{R}^d$, followed by a non-linear real-valued function $\sigma_\star : \mathbb{R} \to \mathbb{R}$. Despite its apparent simplicity, the study of single-index models with Gaussian i.i.d. covariates has been at

the core of the recent theoretical progress in machine learning [TV23; BGJ21; Bie+22; DLS22; BMZ24]. Note that if $\sigma_\star(x) = x$, learning could be achieved efficiently by linear methods such as ridge regression or PCA. The presence of a non-linearity $\sigma_\star$ makes it perhaps the simplest target for which a non-linear hypothesis class is required. Moreover, the choice of isotropic covariates $x \sim \mathcal{N}(0, {}^1\!/_d I_d)$ means that all structure in the target is in learning a representation of σ and a particular direction $w_\star$. The popularity of this model is that different separation results can be shown in the high-dimensional limit where $d \gg 1$:

- In the well-specified setting where we fit an isotropic single-index model with the same hypothesis class (i.e. $f_\theta(x) = \sigma_\star(w^\top x)$), it has been shown that the sample complexity of one-pass SGD[1] is determined by the first non-zero Hermite coefficient of the target $\sigma_\star$, also known as the *information exponent* [BGJ21]. Problems with non-zero first Hermite have information exponent $\ell = 1$, and $w_\star$ can be learned at linear sample complexity $n = O(d)$. Instead, problems with zero first and non-zero second Hermite coefficient have information exponent $\ell = 2$, requiring instead $n = O(d \log d)$ samples [TV23; BGJ21]. Note that this is a factor $\log d$ higher than the Bayes-optimal sample complexity of $n = O(d)$ [Bar+19; Mai+20], hence defining a class of *hard* problems for SGD.

- For fully-connected two-layer neural networks $f_\theta(x) = a^\top \sigma(Wx)$, several results are known under different assumptions. For fixed first layer weights $W \in \mathbb{R}^{p\times d}$ (a.k.a. random features model) and large enough width p, learning the k-th order Hermite coefficient of $\sigma_\star$ requires $n = O(d^k)$ samples [MMM22], implying a sample complexity of $n = O(d^2)$ for a quadratic problem, e.g. $\sigma_\star(x) = x^2$. Recently, it was shown that wide networks ($p \to \infty$) can achieve the well-specified sample complexity of $n = O(d)$ under one-pass SGD, *provided that all Hermite coefficients of both $\sigma_\star$, σ are non-zero* [BMZ24]. In particular, this assumption covers only problems with information exponent $\ell = 1$, excluding hard cases such as quadratic problems. Finally, for $\sigma(x) = \sigma_\star(x) = x^2$, [SVZ20] has shown that for p large enough, full-batch gradient flow achieves sample complexity $n = 2d$, although at a running time of $t = O(\log d)$.

With the exception of [BMZ24], the works mentioned above cover the scaling of the sample complexity in the high-dimensional limit. Our goal is, instead, to derive sharp results for the sample complexity of learning (3.1) with a fully-connected two-layer neural network in the challenging case where $\sigma_\star$ has a vanishing first Hermite coefficient. As discussed above, this case violates the "standard learning scenario" of [BMZ24], and can be seen as a proxy for hard learning problems for descent-based algorithms. For concreteness, in the following we focus on the purely quadratic case:

$$y = \left(w_\star^\top x\right)^2 + \sqrt{\Delta} z, \qquad\qquad w_\star \in \mathbb{S}^{d-1}(\sqrt{d}) \tag{3.2}$$

Learning the target (3.2) consists of learning the non-linearity $\sigma_\star(x) = x^2$ and the direction $w_\star$. In this work, we focus our attention in the second part, considering the following architecture

[1] Which we recall the reader is equivalent to the convergence rate.

with squared-activation:

$$f_\Theta(x) = \frac{1}{p}\sum_{i=1}^{p} a_i (w_i^\top x)^2. \tag{3.3}$$

where $\Theta = (a, W)$ is the set of trainable weights, which are trained with one-pass stochastic gradient descent (SGD):

$$\Theta^{\nu+1} = \Theta^\nu - \gamma \nabla_\Theta \ell(y^\nu, f_{\Theta^\nu}(x^\nu)) \tag{3.4}$$

with square loss $\ell(y, x) = 1/2(y - x)^2$ and initial condition $\Theta^0 = (a^0, W^0)$. Note that at each step ν, the gradient is evaluated at a fresh pair of data $(x^\nu, y^\nu) \in \mathbb{R}^{d+1}$ drawn from the model (3.2). In particular, this implies that after $\nu \in [n]$ steps, the algorithm has seen n data points.

Learning in this problem is hard, and can be compared to finding a needle in a haystack. Indeed, with the exception of one direction that points towards $\pm \boldsymbol{w}_\star$, the population risk at (random) initialization is mostly flat. This slows down the dynamics, which takes a long time to establish a significant correlation with the signal - a scenario we refer to as *escaping mediocrity*.

At first, the particular case of purely quadratic activation might appear too specific. Indeed, as we will see later the population risk for this task has a global maximum at initialization and a degenerate set of global minima. The choice of more general $\sigma_\star$ and σ with zero first Hermite but not necessarily zero higher-order coefficients might give rise to other critical points such as saddle-points, giving rise to a more complex SGD dynamics. However, since the focus of this work is on escaping mediocrity, our conclusions will hold, up to constants, to more general activations with information exponent $\ell = 2$.

Summary of results — Our main contributions in this manuscript are:

- We derive a deterministic set of ODEs providing an exact and analytically tractable description of the one-pass SGD dynamics in the high-dimensional limit $d \to \infty$, and characterize the leading order stochastic corrections to this limit.
- We provide an analytical formula for the number of samples required for one-pass SGD to learn the phase retrieval target in high-dimensions at arbitrary network width. We show that overparameterization can only improve convergence by a constant factor for phase retrieval.
- Finally, we compute the leading order stochastic corrections to the exit time, and show that stochasticity does not help to escape the flat directions at initialization. This suggests that the deterministic description is enough to fully capture the phenomenology of the dynamics in this problem.

All the codes used for numerical experiments are provided in this GitHub repository[2].

[2] `https://github.com/IdePHICS/EscapingMediocrity`

Further related work — The investigation of a deterministic high-dimensional limit of one-pass SGD for two-layer neural networks draws back to the seminal works of [SS95a; SS95b; SS96], and was followed by a stream of works that span decades of research [RB95; CC95; RU98; Gol+19; Gol+20; Gol+22; Vei+22; Arn+23]. More recently, the stochastic corrections around fixed points of the dynamics have been investigated by [BGJ22]. In a complementary research line, [MMN18; CB18; RV22; Gho+19; SS20] have shown that an alternative deterministic description of SGD can be obtained in the infinite width-limit, a.k.a. mean-field regime. High-dimensional reductions of the mean-field equations have been studied by [AAM22; AAM23; HC23; BMZ24]. Recently, [Vei+22; Arn+23] has shown that these apparently different limits of one-pass SGD can be unified in a single description.

There has been a recent surge of interest in studying how increasing degrees of complexity in the target function are incrementally learned by SGD [AAM22; AAM23; Jac+21; BPF22; BMZ24], with an emerging staircase picture where complexity is sequentially learned in different scenarios. This picture, however, is bound to classes of targets where SGD develops strong correlations with the target directions at initialization, a notion which was mathematically formalized by the so-called information exponent ℓ by [BGJ21]. Instead, targets for which the landscape at initialization is mostly flat ($\ell \geq 2$) are hard for SGD at high-dimensions, translating to very slow dynamics. This is precisely the case for the phase retrieval problem ($\ell = 2$), a classic inverse problem arising in many scientific areas, from X-ray crystallography to astronomical imaging [JCH16; Don+23]. Phase retrieval has been widely studied in the literature as a prototypical example of a hard inverse problem [Can+13; CSV13; CLS15; MM18; Bar+19; Aub+20; Mai+20], providing a simple yet challenging example of a non-convex optimization problem which is hard for descent-based algorithms [SQW16; TV23; TV18; Che+19; DDP20; SVZ20; Sar+20; MUZ21].

3.2 High-dimensional limit of SGD

In this section we introduce our key theoretical tool, which consists in low-dimensional reduction of the projected SGD dynamics (3.4) in the high-dimensional limit $d \to \infty$ of interest.

Sufficient statistics — The key observation is to notice that the population risk only depends on the hidden-layer weights $W \in \mathbb{R}^{p\times d}$ through the second layer weights $a \in \mathbb{R}^p$ and the weights' correlation matrices:

$$\Omega \coloneqq \begin{pmatrix} \mathrm{Q} & \mathrm{M} \\ \mathrm{M}^\top & 1 \end{pmatrix} = \begin{pmatrix} {}^1\!/_d \mathrm{W}\mathrm{W}^\top & {}^1\!/_d \mathrm{W}\boldsymbol{w}_\star \\ {}^1\!/_d \left(\mathrm{W}\boldsymbol{w}_\star\right)^\top & 1 \end{pmatrix} \in \mathbb{R}^{(p+1)\times(p+1)}, \tag{3.5}$$

The explicit expression of the population risk is:

$$\begin{aligned} \mathcal{R}(\Theta) =& \mathbb{E}\left[\ell(y, f_\Theta(x))\right] \\ =& \frac{\Delta+3}{2} - \frac{1}{p}\sum_{j=1}^{p} a_j \left(Q_{jj} + 2m_j^2\right) + \frac{1}{2p^2}\sum_{j,l=1}^{p} a_j a_l (Q_{jj}Q_{ll} + 2Q_{jl}^2) \end{aligned} \tag{3.6}$$

Notice that the matrices M, Q are precisely the second moments of the pre-activations $(\lambda_\star, \lambda) = (w_\star^\top x, Wx) \in \mathbb{R}^{p+1}$. Therefore, to characterize the evolution of the risk throughout SGD, it is

sufficient to track the evolution of the second layer weights a_i and the correlation matrices m, Q, which consists of $p(p+1)$ parameters. As shown in Appendix C.1, starting from Equation (3.4) we can derive a set of self-consistent stochastic processes describing the evolution of (a, m, Q):

$$a_j^{\nu+1} - a_j^\nu = \frac{\gamma}{pd}\mathcal{E}^\nu \lambda_j^2 \tag{3.7}$$

$$m_j^{\nu+1} - m_j^\nu = \frac{\gamma}{pd}\,\mathcal{M}_j(a, \lambda_\star, \lambda), \qquad \mathcal{M}_j(a, \lambda_\star, \lambda) \coloneqq 2\mathcal{E}^\nu a_j \lambda_j \lambda_\star \tag{3.8}$$

$$Q_{jl}^{\nu+1} - Q_{jl}^\nu = \frac{\gamma}{pd}\,\mathcal{Q}_{jl}(a, \lambda_\star, \lambda), \qquad \mathcal{Q}_{jl}(a, \lambda_\star, \lambda) \coloneqq 2\mathcal{E}^\nu \left(a_j + a_l\right)\lambda_j\lambda_l + 4\frac{\gamma}{p}\mathcal{E}^{\nu 2}||x^\nu||^2 a_j a_l \lambda_j \lambda_l \tag{3.9}$$

where we have defined the displacement vector

$$\mathcal{E}^\nu \coloneqq (\lambda^{\star\nu})^2 - \frac{1}{p}\sum_{j=1}^{p} a_j (\lambda_j^\nu)^2 + \sqrt{\Delta} z^\nu, \tag{3.10}$$

and we used γ/d as the learning rate of the second layer, in order to have the same high-dimensional scaling.

High-dimensional limit — As of now we have not made any assumptions on the dimension of the problem; the stochastic processes defined in (3.7) are exact, with the right-hand side depending implicitly on (m, Q) through the moments of $(\lambda_\star, \lambda)$. However, our goal is to study this process in the high-dimensional limit $d \to \infty$ where learning is hard and simulating (3.4) can be computationally demanding. Defining a step-size $\delta t = \gamma/pd$ and a continuous extension of (a^ν, m^ν, Q^ν) to continuous time $(a(\nu\delta t), m(\nu\delta t), Q(\nu\delta t))$ by linear interpolation, it can be shown that in the high-dimensional limit $d \to \infty$ the sufficient statistics $(a(t), m(t), Q(t))$ concentrate in their expectation $(\bar{a}(t), \bar{m}(t), \bar{Q}(t))$, which satisfies the following system of ordinary differential equations (ODEs):

$$\begin{aligned}
\frac{\mathrm{d}\bar{a}_j}{\mathrm{d}t} &= \mathbb{E}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{E}\lambda_j^2\right] \\
\frac{\mathrm{d}\bar{m}_j}{\mathrm{d}t} &= \mathbb{E}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{M}_j(a, \lambda_\star, \lambda)\right] =: \Psi_j\left(\Omega\right) \\
\frac{\mathrm{d}\bar{Q}_{jl}}{\mathrm{d}t} &= \mathbb{E}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{Q}_{jl}(a, \lambda_\star, \lambda)\right] =: \Phi_{jl}\left(\Omega\right)
\end{aligned} \tag{3.11}$$

with initial conditions given by $(\bar{a}(0), \bar{m}(0), \bar{Q}(0)) = (a^0, {}^1\!/_d W^0 w_\star, {}^1\!/_d W^0 W^{0\top})$. The explicit expression of these expected values can be found in Appendix C.1. As discussed in the related works, the high-dimensional limit of one-pass SGD for two-layer neural networks have been studied under different settings in the literature [SS96; Gol+19; RU98; TV18; Vei+22; Arn+23; BGJ22; BMZ24]. However, to our best knowledge our work is the first to derive and study these equations for the squared activation in the high-dimensional limit.

Initialization and mediocrity — In the noiseless case $\Delta = 0$, it is easy to check that $a_j = 1$ and $w_j = \pm w_\star$ ($m_j = \pm 1$ and $Q_{jl} = 1$) is indeed a stationary point of (3.11) that corresponds to two degenerated global minima of the population risk (3.6). Adding a noise $\Delta > 0$ only shift these values. Similarly, it is easy to check that $m_i = 0$ and $\mathrm{Q}_{ij} = 0$ for $i \neq j$ are also stationary points. These correspond to taking $w_j \perp w_l \perp w_\star$ for all $j \neq l$ in (3.11), and is a global maximum of (3.6). This stationary point plays an important role in the dynamics. Indeed, in the absence of knowledge on the process that generated the data (3.2), it is customary to initialize the weights randomly:

$$w_j^0 \sim \mathcal{N}(0, I_d), \qquad\qquad j = 1, \cdots, p. \tag{3.12}$$

When $d \to \infty$, the weights are being orthogonal with high probability. In terms of the sufficient statistics:

$$\mathrm{Q}_{jj} \sim \mathrm{Dirac}(1), \qquad j \neq l : \sqrt{d}\mathrm{Q}_{jl}^0 \xrightarrow{d\to+\infty} \mathcal{N}(0,1) \quad \text{and} \quad \sqrt{d}\mathrm{M}_j^0 \xrightarrow{d\to+\infty} \mathcal{N}(0,1). \tag{3.13}$$

Therefore, since the variance of (m^0, Q^0) decays as $1/d$, the higher the dimension, the closer a random initialization is to a stationary point of the dynamics. Moreover, of all the d directions, there exists $d-p-1$ directions orthogonal to $w_\star$ and $\{w_j^0\}_{j\in[p]}$ along which the population risk (3.6) remains constant. The proliferation of flat directions close to initialization severely slows down the SGD dynamics at high-dimensions, which typically requires $n = O(d \log d)$ steps to develop a significant correlation with the signal in order to escape this region. This scenario, which we refer to as *escaping mediocrity*, is common to many hard learning problems [BGJ21]. In the following, we leverage the exact description (3.11) derived in this section to estimate precisely how much data is required for SGD to escape mediocrity in the prototypical phase retrieval problem (3.1).

Spherical constraint — A phenomenon that is observed when starting from the initial conditions above is a change in the norms of the weights w_i without effectively correlating with $\boldsymbol{w}_\star$. In this phase, sometimes referred as *norm learning*, $m \approx Q_{jl} \approx 0$ for $j \neq l$, while Q_{jj} changes considerably, resulting in a slight decrease in the population risk towards a plateau that reflects mediocrity. Since the focus of this study is precisely on escaping mediocrity (i.e. developing non-zero correlation with the signal), in the following we will fix the norm of the weights $w_i^\nu \in \mathbb{S}^{d-1}(\sqrt{d})$ at initialization and throughout the dynamics $\nu \in [n]$. This assumption, which was also the focus of [BGJ21], amounts to imposing a spherical constraint at every step of SGD, also known as *projected SGD*:

$$\boldsymbol{w}_j^{\nu+1} = \frac{\boldsymbol{w}_j^\nu - \gamma \nabla_{\boldsymbol{w}_j} \ell(y^\nu, f_{\Theta^\nu}(x^\nu))}{\left\| \boldsymbol{w}_j^\nu - \gamma \nabla_{\boldsymbol{w}_j} \ell(y^\nu, f_{\Theta^\nu}(x^\nu)) \right\|} \sqrt{d}. \tag{3.14}$$

The high-dimensional limit of these equations lead to the following ODEs for the evolution of the sufficient statistics (M, Q):

$$\frac{\mathrm{d}\bar{m}_j}{\mathrm{d}t} = \Psi_j(\Omega) - \frac{\bar{m}_j}{2}\Phi_{jj}(\Omega), \qquad \frac{\mathrm{d}\bar{\mathrm{Q}}_{jl}}{\mathrm{d}t} = \Phi_{jl}(\Omega) - \frac{\bar{\mathrm{Q}}_{jl}}{2}\left(\Phi_{jj}(\Omega) + \Phi_{ll}(\Omega)\right). \tag{3.15}$$

Note that $\mathrm{Q}_{jj} = 1$ is consistently fixed.

3.3 Escaping mediocrity in the well-specified scenario

As a starting point, we consider the well-specified case of $p = 1$. As we are going to see, this case captures almost all the phenomenology of interest, and is helpful to build an intuition for how SGD escapes mediocrity. Moreover, despite its apparent simplicity, this case has been the subject of several works in the literature [SQW16; TV23; Che+19; BGJ21]. In particular, [Che+19] established a convergence rate $t = O(\log d)$ for randomly initialized gradient descent on the population risk. In our setting (3.4), this corresponds to the $\gamma \to 0^+$ limit, and translates to a sample complexity of $n = O(d \log d)$. [TV23] provided a refined analysis that accounts for the stochasticity in one-pass SGD, reaching a similar $n = O(d \log d)$ rate for escaping mediocrity, in agreement with the general information exponent characterization of [BGJ21].

In this section, we show that in the high-dimensional limit, the sample complexity constant for one-pass SGD can be well estimated from the deterministic reduction (3.7). In particular, we show that the stochastic corrections from a finer analysis of the process (3.7) can be neglected.

3.3.1 Population risk landscape

As previously discussed, one-pass stochastic gradient descent can be seen as a discretization of gradient flow on the population risk [RM51]. Therefore, understanding the geometry of the population risk can provide useful insight into the behavior of SGD. For $p = 1$ and imposing the spherical constraint, the population risk (3.6) considerably simplifies since everything can be expressed in terms of a single, scalar sufficient statistics $m = {}^1\!/\!_d w_\star^\top w \in [-1, 1]$:

$$\mathcal{R}(w) - \frac{\Delta}{2} = 2(1 - m^2) \tag{3.16}$$

From this expression, it is easy to see that we have two global minima $m = \pm 1$ (corresponding to $w = \pm w_\star$) and one global maximum $m = 0$ (corresponding to $w \perp w_\star$). Indeed, the spherical gradient of the population risk can also be computed explicitly (see Appendix C.7.1 for details):

$$\operatorname{grad}_{\mathbb{S}^{d-1}} \mathcal{R}(w) = 4m\,(mw - w_\star) \tag{3.17}$$

which confirm that these are the only critical points. Finally, we can also compute the spherical Hessian exactly:

$$\operatorname{Hess}_{\mathcal{S}^{d-1}} \mathcal{R}(w) = 4\left[m^2 I_d + m w w_\star^\top - w_\star w_\star^\top\right] \tag{3.18}$$

Note that for $w = \pm w_\star$ ($m = \pm 1$) the Hessian is proportional to the identity, hence positive definite as expected for a minimum. Instead, for $w \perp w_\star$ ($m = 0$), the Hessian becomes a rank-one matrix with $d - 1$ zero eigenvalues and a single negative eigenvalue with eigenvector proportional to $w_\star$. Hence, $m = 0$ also corresponds to a *strict saddle*, i.e. the landscape is flat along most of the directions, except for one that points towards the signal. Note that with high probability, we have precisely $m \approx 0$ for an uninformed random initialization in high-dimensions $d \to \infty$.

This provides a typical picture of mediocrity in high-dimensions, where the landscape resembles a flat golf course with a single hole, see Figure C.3 in Appendix C.7.

3.3.2 Exit time from deterministic limit

We now move to the description of the one-pass SGD dynamics. Our key goal in this section is to determine how much data / how long SGD takes in order to find the signal in the high-dimensional limit $d \to \infty$. As we have discussed in Section 3.2, in this limit the sufficient statistics concentrate, with its evolution being described by the following deterministic ODE:

$$\frac{\mathrm{d}\bar{m}(t)}{\mathrm{d}t} = \bar{m}(t)\left[4(1-6\gamma)(1-\bar{m}^2(t)) - 2\gamma\Delta\right] \quad \text{with} \quad \bar{m}(t) \in [-1,1] \tag{3.19}$$

with initial condition $\bar{m}(0) = {}^1\!/_d w_\star^\top w^0$. See Appendix C.2 for an explicit derivation. Figure 3.1 (left) compares the evolution of the risk predicted from solving the high-dimensional ODEs (3.19) with different finite size ($d = 3000$) simulated instances of the problem, showing a good agreement between the theory and the averaged population risk over the different runs.

Given the spherical constraint, the population risk is now simply given by $\mathcal{R}(m) = 2\left(1-m^2\right) + {}^\Delta\!/_2$. From this expression, it is clear that $m = \pm 1$ are global minima and $m = 0$ is a global maximum. Therefore, the information theoretically minimum achievable risk is $\mathcal{R}(\pm 1) = \min \mathcal{R}(m) = {}^\Delta\!/_2$.

We start with two immediate observations that can be drawn from (3.19). First, we have a necessary upper bound on the learning rate for learning to occur: $\gamma < {}^1\!/_6$. Moreover, from fixed-point stability analysis we can get the value where $\bar{m}$ converges for large times, and, consequently, the asymptotic excess population risk achievable in this setting is:

$$\lim_{t\to\infty} \mathcal{R}(\bar{m}(t)) - {}^\Delta\!/_2 = \frac{\gamma\Delta}{1-6\gamma}. \tag{3.20}$$

The presence of an asymptotic risk plateau for the excess risk is consistent with previous results for the high-dimensional limit of two-layer neural networks [SS96; Gol+19]. Note that in the population gradient flow limit $\gamma \to 0^+$, SGD converges to the minimal risk [RM51]. Indeed, the presence of a finite excess risk even in the well-specified setting is an intrinsic correction from the SGD noise in the high-dimensional regime where ${}^1\!/_d \ll \gamma$ [Vei+22], and can be related to the radius of the asymptotic stationary distribution of the weights around the global minima [Pfl86; DDB20].

We now move to our main problem: estimating the time SGD takes to escape mediocrity at initialization. Let $T \in [0,1]$ be the relative difference with respect to the initial value of the risk, and let t_{ext} be the time when the risk exits the region above the threshold T, see Figure 3.1 (right) for an illustration. By construction, t_{ext} can be found by solving the following equation:

$$(1-T)\left(\mathcal{R}\big(\bar{m}(0)\big) - \frac{\Delta}{2}\right) = \mathcal{R}\left(\bar{m}\left(t_{\text{ext}}\right)\right) - \frac{\Delta}{2}. \tag{3.21}$$

The above can be exactly solved by numerically integrating (3.19) and then finding the root of (3.21). However, an analytical expression for the ODE exit time can be found from the following two observations:

- From the discussion around Equation (3.13), initializing at random in high-dimensions imply that $\bar{m}(0) = \varepsilon \ll 1$, so we can consider the linearization of equation (3.19) in ε and solve it analytically. For small enough T, this will lead us to an accurate result;
- Even if the ODE trajectories are deterministic, the exit time t_{ext} is a random variable of the random initialization.

Note that these lead to two natural notions of average exit time over the initial conditions. The first one is obtained by taking the expected value over initial conditions before solving the cross-threshold equation

$$t_{\text{ext}}^{(\text{anl})} = \frac{\log\left[Td + (1-T)\right]}{8(1-6\gamma) - 4\gamma\Delta}. \tag{3.22}$$

Borrowing the jargon from statistical physics, we refer to $t_{\text{ext}}^{(\text{anl})}$ as the *annealed exit time*. The second option is to take the expected value exit time obtained from solving (3.19) over a fixed initial condition:

$$t_{\text{ext}}^{(\text{qnc})} = \mathbb{E}_{\mu_0 \sim \chi^2(1)} \left[\frac{\log\left[\frac{Td}{\mu_0} + (1-T)\right]}{8(1-6\gamma) - 4\gamma\Delta} \right]. \tag{3.23}$$

Again, borrowing the jargon from statistical physics we refer to $t_{\text{ext}}^{(\text{qnc})}$ as the *quenched exit time*. Some comments on this result are in order:

- By concavity of the logarithm function, we have $t_{\text{ext}}^{(\text{qnc})} \geq t_{\text{ext}}^{(\text{anl})}$.
- For both notions, we have $t_{\text{ext}} = O(\log d)$ implying $n = O(d \log d)$ samples are required to escape mediocrity, consistent with the rates in the literature [Che+19; TV23; BGJ21].
- Both exit times are monotonically increasing in both $\gamma \in [0, 1/6]$ and $\Delta \geq 0$. Recalling that $\delta t = \gamma/d$, this implies the existence of an optimal learning rate $\gamma_{\text{opt}} = 1/(12 + \Delta)$ that minimizes the number of samples required to escape mediocrity.

3.3.3 Does stochasticity matter?

Note that the initial correlation parameter at random initialization (3.12) is given by:

$$m^0 = O(1/\sqrt{d}) \tag{3.24}$$

Therefore, in the high-dimensional limit $d \to \infty$ in which the ODE description (3.19) is exact, we have $\bar{m}(0) = 0$. This is a fixed point (3.19), which suggests that strictly in the high-dimensional limit SGD is trapped forever at mediocrity. However, in practice we always have $d < \infty$, meaning that at initialization we always have a non-zero correlation with the signal $m^0 = \varepsilon \ll 1$. Moreover, at high but finite dimensions, (3.19) is just an approximation to the actual stochastic dynamics (3.7). Indeed, this is precisely what we used in order to estimate the exit time from the deterministic ODE (3.19). While the stochastic corrections to the high-dimensional limit do not radically change the convergence rate scaling [TV23] (and hence the mediocrity picture), it is important to ask whether it leads to important corrections on the precise exit time.

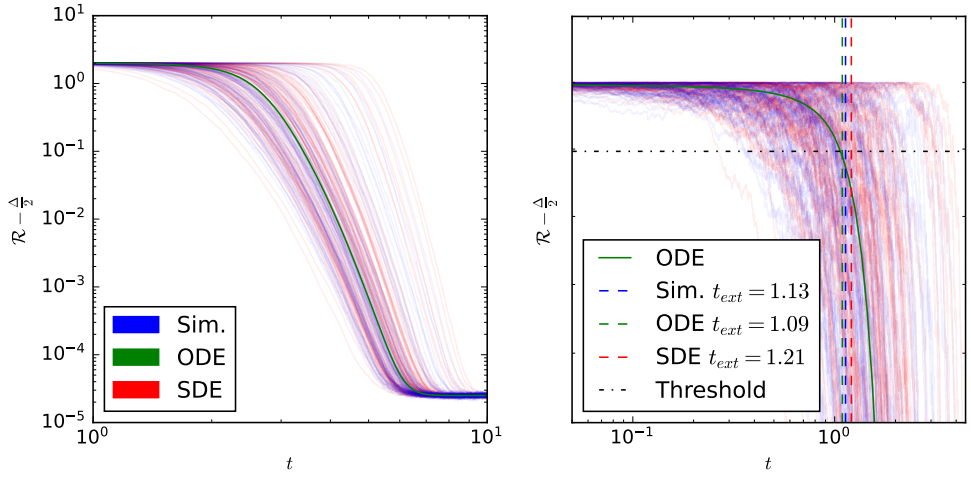


Figure 3.1: Multiple run of the simulated SGD and the numerically integrated SDE, always starting from the same initial condition, with $d = 3000$. All the t_{ext} presented are obtained by solving numerically (3.21). The SDE captures the variance that the ODE doesn't exhibit, but the t_{ext} do not change considerably.

Stochastic corrections to the deterministic high-dimensional limit of one-pass SGD have been recently discussed in a broad setting by [BGJ22]. In particular, this work has shown that close to a fixed point the process for the sufficient statistics (3.7) can be well approximated in the high-dimensional limit by a diffusion process with drift potential given by the corresponding deterministic ODEs. We follow a similar strategy, and consider the following process specialized in the case $p = 1$:

$$\begin{aligned} \mathrm{dM}_1 &= \Psi_1(\Omega)\,\mathrm{d}t + \sqrt{\frac{\gamma}{d}}\boldsymbol{\sigma}_{\mathrm{M}}(\Omega)\cdot \mathrm{d}B_t \\ \mathrm{dQ}_{11} &= \Phi_{11}(\Omega)\,\mathrm{d}t + \sqrt{\frac{\gamma}{d}}\boldsymbol{\sigma}_{\mathrm{Q}}(\Omega)\cdot \mathrm{d}B_t \end{aligned} \tag{3.25}$$

where $\mathrm{d}B_t$ is a 2-dimensional Wiener process, and $\boldsymbol{\sigma}_M$ and $\boldsymbol{\sigma}_Q$ are defined as

$$\begin{pmatrix} \boldsymbol{\sigma}_{\mathrm{M}} \\ \boldsymbol{\sigma}_{\mathrm{Q}} \end{pmatrix} := \sqrt{\begin{pmatrix} \mathrm{Var}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{M}_1\right] & \mathrm{Cov}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{M}_1,\mathcal{Q}_{11}\right] \\ \mathrm{Cov}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{M}_1,\mathcal{Q}_{11}\right] & \mathrm{Var}_{(\lambda,\lambda_\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\mathcal{Q}_{11}\right] \end{pmatrix}}. \tag{3.26}$$

The stochastic corrections are analogous with the ones obtained by [BGJ22]; in Appendix C.6 we present an alternative derivation of them, following their approach, and we highlight the fact that our formulae generalize their result. Notice that the stochastic correction is proportional to $\sqrt{\gamma/d}$, consistent to a first order correction to the deterministic limit. Similarly to the discussion in Section 3.3.2, the spherical constraint can be imposed by projecting the process in the sphere. This is discussed in detail in Appendix C.2, and for $p = 1$ reads:

$$\frac{\mathrm{dM}_1}{\mathrm{d}t} = \left(\Psi_1(\Omega) - \frac{\mathrm{M}_1}{2}\Phi_{11}(\Omega)\right)\mathrm{d}t + \sqrt{\frac{\gamma}{d}}\left(\boldsymbol{\sigma}_{\mathrm{M}} - \frac{m_1}{2}\boldsymbol{\sigma}_{\mathrm{Q}}\right)\cdot \mathrm{d}B_t \tag{3.27}$$

Figure 3.1 compares different instances of finite size simulations with instances of the SDE (3.27) with the same initial condition. Although the stochastic correction offers a better description of the process at large but finite dimensions, we find that quite surprisingly they have a small impact in the exit time. Hence, the formulas (3.22) & (3.23) derived in the Section 3.3.2 for random initialization provide a good approximation to the exit time. In Appendix C.4 we discuss how to derive an exit time formulae with the stochastic corrections, both annealed and quenched one. As just shown, the new formulas do not offer any improvements compared to the deterministic ones, nevertheless the stochastic process can describe the dynamics even when the initialization is exactly $m = 0$, that is a fixed point of the ODE.

To summarize, in this section we have shown that the deterministic ODEs provides a good approximation for the precise number of samples required for escaping mediocrity in high-dimensions. In other words, stochasticity *does not help* in navigating the flat directions at initialization and correlating with the signal.

3.4 The role of width

Thus far our discussion has focused on the well-specified case. We now discuss the role of width in escaping mediocrity. Our starting point are the deterministic ODEs (3.11) for the sufficient statistics derived in Section 3.2. As in our previous analysis, we focus on the spherical setting where $w_i \in \mathbb{S}^{d-1}(\sqrt{d})$, implying $Q_{jj} = 1$, see Appendix C.2 for a detailed derivation. First, we derive analytical expressions for the exit time for arbitrary width $p \geq 1$ in the particular case where the second layer is fixed at initialization $a_j^0 = 1, \forall j \in [p]$. The role played by the second layer is then discussed in Subsection 3.4.1.

Differently from the $p = 1$ case, the process cannot be described by a single sufficient statistic, and instead we have to track $p(p-1)/2$ non-diagonal entries of Q (it is a symmetric matrix), and p components of the vector M. Note that equation (3.21) remains valid to define t_{ext}, and can be solved numerically. An analytical expression for the exit time can be derived under similar assumptions to the ones discussed in Section 3.3.2, although the derivation is significantly more cumbersome. Full details can be found in Appendix C.3. The final expressions for the annealed and quenched exit times are given by

$$t_{\text{ext}}^{(\text{anl})} = \frac{\log\left[\frac{T(p+1)d+(p+1)(1-T)}{2p}\right]}{8\left[1-\frac{\gamma}{p}\left(1+\frac{1}{p}+\frac{4}{p^2}+\frac{\Delta}{2}\right)\right]}, \tag{3.28}$$

$$t_{\text{ext}}^{(\text{qnc})} = \mathbb{E}_{\mu_0,\tau_0\sim\mathcal{P}_p^d}\left[\frac{\log\left[\frac{Tp(p+1)d+(2\mu_0 p-\tau_0)(1-T)}{2\mu_0 p}\right]}{8\left[1-\frac{\gamma}{p}\left(1+\frac{1}{p}+\frac{4}{p^2}+\frac{\Delta}{2}\right)\right]}\right]. \tag{3.29}$$

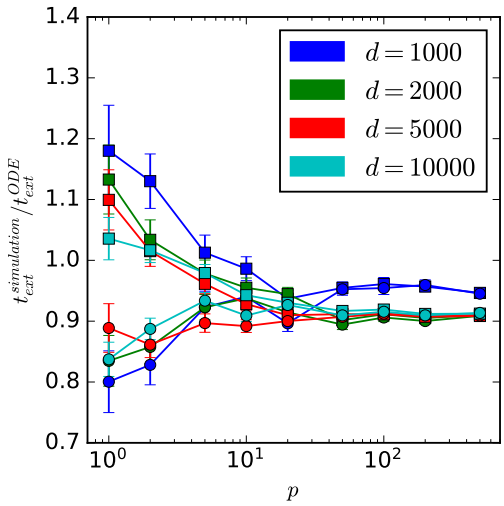


Figure 3.2: Ratio between the measured t_{ext} from simulations and the corresponding analytical formula (square = annealed, circle = quenched). We average over many initial conditions, for different values of p. The ratio γ/p has been kept constant for different simulations.

With the distribution of the variables τ_0 and μ_0 given by

$$\mu_0, \tau_0 \sim \mathcal{P}_p^d \quad \text{where } \mathcal{P}_p^d \equiv \left(d\sum_{j=1}^{p}(u_j \cdot v)^2, 2d\sum_{j=1}^{p}\sum_{l=j+1}^{p}(u_j \cdot u_l)^2 \right) \text{ with } v, u_j \sim \mathbb{S}^{d-1}(1).$$

Notice that t_{ext} is a monotonically decreasing function of the width. Nevertheless, for any $p \geq 1$, the leading order dependence in the dimension is $t_{\text{ext}} = \log d$. Hence, despite helping to escape mediocrity, increasing the width cannot mitigate it. This can be contrasted to other aspects in which overparameterization can significantly help optimization, for instance with global convergence [BC22]. Interestingly, the minimal escaping time

$$t_{\text{ext}}^{(\text{anl})} = \frac{1}{4}\log\left(\frac{T(p+1)d + (p+1)(1-T)}{2p}\right)$$

, obtained by choosing the learning rate that minimizes the sample complexity for escaping, has the same pre-factor for any width $p \geq 1$, with the only differences being the dependence in p inside the logarithm and in the time scaling $t = \nu\gamma/pd$. At infinite width $p \to \infty$, this simply amounts to a factor $\frac{12+\Delta}{2+\Delta}$ with respect to $p = 1$. Details of this computation can be found in Appendix C.3.4.

Figure 3.2 compares our analytical formulas (3.28) and (3.29) with real one-pass SGD simulations. The simulation are averaged over many different instances of the initial conditions, and the ratio γ/p is kept constant when varying p, for not having discrepancies due to the different learning rate scaling. It's interesting to notice how the two different formulas give the same outcome for large width $p \gg 1$. Moreover, for narrow networks they essentially differ by a d-independent constant.

Figure 3.2 also suggests that, as for $p = 1$, the stochasticity can be neglected in the estimation of the exit time. In Appendix C.5 we provide further evidence of that.

3.4.1 Training the second layer

In the previous section, we have derived analytical expressions for the exit time in the particular case of fixed second layer weights $a_j^0 = 1$. Here, we provide numerical evidence that training the second layer does not significantly change our conclusions.

The key challenge is that by training the second layer we can't measure t_{ext} as the time needed to escape the risk at initialization. Indeed, from equation (3.21) it can be seen that in the very first steps of the learning the vector a changes slightly to adapt to the initial conditions, thereby fitting the noise. In this scenario, instead of looking directly at the risk, we can instead use the largest component of the correlation vector m as a measure on how much the network has learned. At random initialization, this is of order $1/\sqrt{d}$ and grows to 1 as the neural network correlates with the target weights. A natural choice for initializing the second layer weights is $a_j^0 = 1, \forall j \in [p]$. In principle, this initial condition guarantees that the risk at initialization is exactly equal to the case where a_j is fixed. On the other hand, as we already point out, the initial plateau where the dynamics gets stuck depends on the particular second layer initial condition. Even for other choices of initialization, e.g. $a_j \sim \text{Bernoulli}(1/2)$, the dynamics quickly goes to a plateau, so it does not really matter which a_j^0 is used. Therefore, for simplicity we choose a homogeneous initialization $a_j^0 = 1$. Figure 3.3 compares the evolution of the maximum correlation when learning the second layer or not, for different values of p.

It is important to stress that we are not claiming that the time needed to reach the minimum of the population risk is the same when training or not the second layer, as can be seen in Figure 3.3. Instead, our result highlights that the time needed to escape the flat directions at initialization are close. In fact, after the two layer neural network has escaped mediocrity, the dynamics can be very different whether the second layer weights are trained or not. For instance, a could become sparse with just a few neurons contributing to the output, or it could remain close to homogeneous $a_j = 1$, and with all neurons correlating with the target. Although studying the dynamics after escaping mediocrity is surely an interesting endeavor, it's out of the scope of this manuscript.

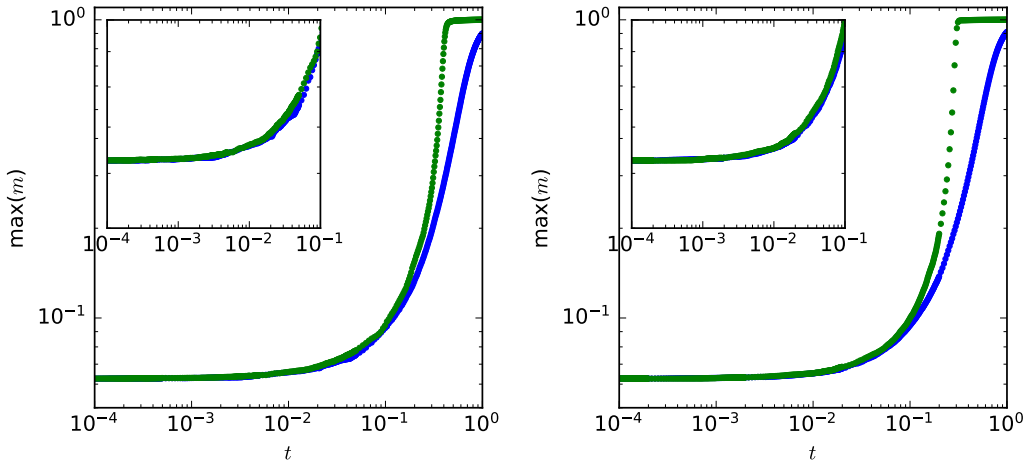


Figure 3.3: $p = 20$ (left), $p = 50$ (right), $d = 1000$. Comparison between the growth of max$m$ throughout the learning process, when the second layer is fixed (blue) and trained (green). The dynamics is obviously different far from the starting point, but when we zoom close to the exit point, the two processes have the same behavior, t_{ext} included.

3.5 Conclusion

In this work we have derived a sharp formula for how many samples are required for a two-layer neural network trained with one-pass SGD to learn a quadratic target in the high-dimensional limit. In particular, we have shown that increasing the width can only improve the sample complexity by a pre-factor, with the overall scaling with the dimension remaining of the same order $n = O(d \log d)$. Therefore, for this target overparameterization does not significantly help optimization, providing a prototypical example of a hard class of functions for learning with SGD. Our results rely on a low-dimensional description of SGD in terms of a stochastic process describing the evolution of the sufficient statistics in the high-dimensional limit. Surprisingly, we have shown that deriving the sample complexities from the deterministic drift of this process with small initial correlation with the target provides a fairly good approximation of the exit time as computed from the full process with zero initial correlation, showing stochasticity does not play a crucial role in escaping mediocrity for this problem.

Acknowledgements

We thank Gérard Ben Arous and Lenka Zdeborová for valuable discussions. LA would like to thank *Scuola Normale Superiore* and *Università di Pisa* for the support during part of this project. We acknowledge funding from the Swiss National Science Foundation grant SNFS OperaGOST, 200021_200390 and the *Choose France - CNRS AI Rising Talents* program.

Chapter 4

Online Learning and Information Exponents: **On The Importance of Batch size, and Time / Complexity Tradeoffs**

Luca Arnaboldi, Yatin Dandi, Florent Krzakala, Bruno Loureiro, Luca Pesce, and *Ludovic Stephan*

Abstract. We study the impact of the batch size n_b on the iteration time T of training two-layer neural networks with one-pass stochastic gradient descent (SGD) on multi-index target functions of isotropic covariates. We characterize the optimal batch size minimizing the iteration time as a function of the hardness of the target, as characterized by the information exponents. We show that performing gradient updates with large batches $n_b \lesssim d^{\ell/2}$ minimizes the training time without changing the total sample complexity, where ℓ is the information exponent of the target to be learned [BGJ21] and d is the input dimension. However, larger batch sizes than $n_b \gg d^{\ell/2}$ are detrimental for improving the time complexity of SGD. We provably overcome this fundamental limitation via a different training protocol, *Correlation loss SGD*, which suppresses the self-interaction term in the loss function. We show that one can track the training progress by a system of low-dimensional ordinary differential equations (ODEs). Finally, we validate our theoretical results with numerical experiments.

4.1 Introduction

Descent-based algorithms, such as Stochastic Gradient Descent (SGD) and its variations, are the backbone of contemporary machine learning. Their simplicity in implementation, efficiency in operation, and notably effective performance in practice highlight their importance. A mathematical understanding of SGD's effectiveness remains a key focus in the field. Recent progress

has been particularly noteworthy in the realm of shallow neural networks. A sequence of works demonstrated that optimizing large width two-layer neural networks can be mapped into a convex optimization problem over the space of probability measures of weights, the so-called mean-field analysis [MMN18; CB18; RV22; SS20]. Following this breakthrough, a large part of the theoretical effort has shifted to describing what class of functions can be efficiently learned by SGD, i.e. time and computational complexities required to learn a given class of functions. This has been, in particular, thoroughly analyzed in a series of recent works focusing on isotropic distributions (e.g. Gaussian, spherical or in the hypercube) and targets depending only on a few relevant directions (a.k.a. *multi-index models*). A key result from this literature is that the time complexity of SGD scales with the covariates dimension according to the so-called *information exponent* [BGJ21] for single-index and *leap complexity* [Abb+21; AAM23] for multi-index targets, sparking increasing interest from the theoretical machine learning community over the last few months [DLS22; Dam+23; Dan+24a; BBP23; Ba+23; Mon+24; Mou+23; ZB24].

Paper written in Winter 2024.

Our work follows this thread, focusing instead on the effect of batch size n_b, parallelization, and sample-splitting into the overall complexity required to learn a multi-index target. Instead of looking at data one-by-one, as is common in theoretical studies, we investigate the finite n_b problem, and characterize the time/complexity tradeoff when learning with one-pass SGD. Our central goal is to paint a complete picture of how fast generalized linear models and two-layer neural networks adapt to the features of training data as a function of n_b, and the structure of the target function.

Our analysis sheds light on a fundamental limitation of one-pass (or online) SGD, namely that for batch sizes larger than the input dimension, the dynamics of the training algorithm is dominated by negative feedback terms that do not permit to reduce the time iterations needed to learn the target. Therefore, we provide a rigorous solution to this fundamental limitation of SGD by considering gradient updates on the correlation loss. Our approach, drawing inspiration from the *summary statistics* method employed by [SS95b; BGJ21; BGJ22], concentrates on the overlaps of neurons with the target subspace and their norms. This differs from recent studies, such as those by [AAM22] and [DLS22], which focus on the full gradient vector.

4.2 Setting, contributions, and related works

Consider a two-layer neural network with activation function σ and first and second layer weights given by $W \in \mathbb{R}^{p\times d}$ and $\boldsymbol{a} \in \mathbb{R}^p$ respectively:

$$f(\boldsymbol{z}) = \frac{1}{p}\sum_{j=1}^{p} a_j \sigma(\langle \boldsymbol{z}, \boldsymbol{w}_j\rangle). \tag{4.1}$$

We are interested in studying the capacity of f to learn from training data $\mathcal{D} = \{(\boldsymbol{z}^\nu, y^\nu)_{\nu\in[N]} \in \mathbb{R}^{d+1}\}$. In the following, we work under the following setting.

In this chapter the input covariate is denoted $\boldsymbol{z}$, playing the role of $\boldsymbol{x}$ in the rest of the thesis.

Data model — As hinted in the introduction, we focus on the case the (noisy) labels depend on the covariates only through a projection over a k-dimensional subspace:

$$y^\nu = h^\star(W^\star \boldsymbol{z}^\nu) + \sqrt{\Delta}\xi^\nu, \quad \boldsymbol{z}^\nu \sim \mathcal{N}(0, I_d) \tag{4.2}$$

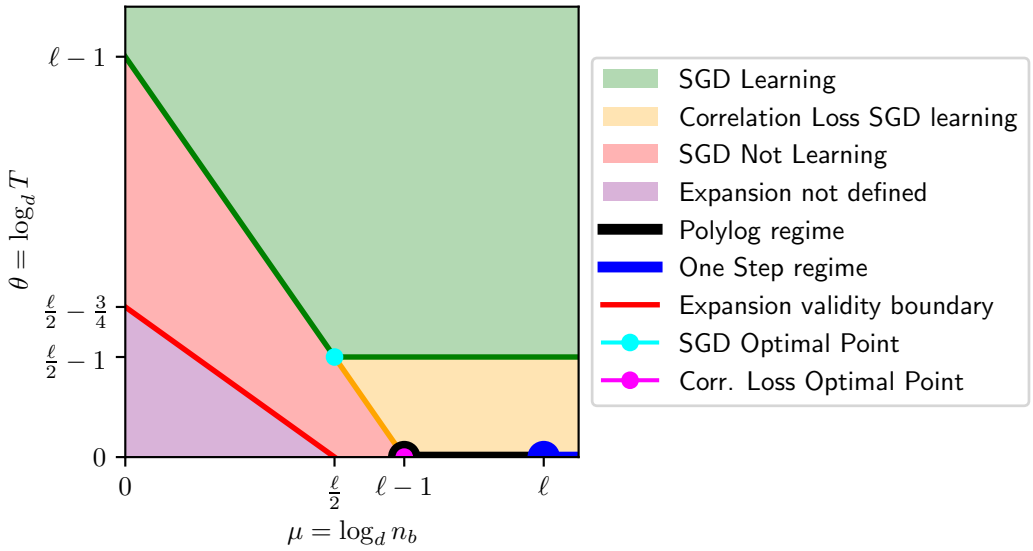


Figure 4.1: **Time / Batch size tradeoff for weak recovery:** Phase diagram illustrating different SGD learning regimes as a function of the batch size exponent $\mu = \log_d n_b$ and weak recovery time exponent $\theta = \log_d T$. The analysis is dependent on the target's information exponent ℓ, this particular plot is valid when $\ell \geq 3$. **Not correlating region:** SGD is not able to achieve weak recovery. **Self-interaction regime:** SGD is not able to perform weak recovery, but Correlation loss SGD overcomes this limitation. **Weak recovery region:** SGD successfully achieve weak recovery. Note that it exists an **optimal choice** at batch size $n_b = O(d^{\ell/2})$ that minimizes the number of iterations needed by SGD, and another **optimal point** at $n_b = O(d^{\ell-1})$ for *Correlation Loss*. The critical line where $n_b = \Omega(d^{\ell-1})$ is not addressed by our formal. See details about the other two regions (**Poly-log Regime** and **One-step regime** [Dan+24a]) in Appendix D.4.

where $W^\star = \{\boldsymbol{w}_r^\star\}_{r\in[k]} \in \mathbb{R}^{k\times d}$ are the target weights, $h^\star : \mathbb{R}^k \to \mathbb{R}$ is a non-linear activation function, $\xi^\nu \sim \mathcal{N}(0,1)$ is the label noise with variance given by $\Delta \geq 0$. We focus on the case where $k = O(1)$ and d is large, i.e. the label only depends on a few directions of a high-dimensional ambient space. The target function $f^\star(\boldsymbol{z}) = h^\star(W^\star\boldsymbol{z})$ is often referred to in the literature as a *multi-index model*.

Note that the setting above where we assume a generative model for the data and study the capacity of a model to learn is also known as teacher-student model in the literature. We adopt this terminology and refer to $f^\star$ and f as the *teacher* and the *student* functions, respectively. Similarly, we refer to $W^\star$ and W as the *teacher* and *student* weights.

Hardness of the learning task — Characterizing what class of targets are efficiently learned by two-layer networks is arguably one of the key questions in theoretical machine learning. The pivotal work of [BGJ21] provably describes that for $k = 1$, the hardness of the learning task is encoded by a single number, the information exponent ℓ. More precisely, given the activation $h^\star$ in (4.2), ℓ is the lowest degree of the Hermite polynomials $\{\mathrm{He}_j\}_{j\in\mathbb{N}}$ appearing in the Hermite expansion of $h^\star$. This notion generalizes direction-wise for multi-index models ($k > 1$), where ℓ is known as leap complexity [AAM23].

Definition 4.1 (Information Exponent [BGJ21])**.**

$$\ell = \min\{j \in \mathbb{N} : \mathbb{E}_{\xi\sim\mathcal{N}(0,1)}\left[h^\star(\xi)\mathrm{He}_j(\xi)\right] \neq 0\} \tag{4.3}$$

Training algorithm — Given the training data $\mathcal{D}$, we consider the training of $(W, \boldsymbol{a})$ under a *sample splitting* scheme: the data is partitioned $\mathcal{D} = \bigcup_{t=1}^T \mathcal{D}_t$ into $T = \lfloor N/n_b \rfloor$ disjoint batches $\mathcal{D}_t$ of size n_b, which are used, one every iteration, to train the network. We consider a common assumption for the training algorithm that is to decouple the training of the hidden weights W and the second layer weights $\boldsymbol{a}$. By keeping fixed the second layer weights at initialization $\boldsymbol{a} = \boldsymbol{a}_0$, the hidden layer weights W are estimated using (projected) SGD:

This chapter departs from the discrete/continuous time convention of Section 1.2: here t denotes the discrete SGD step (called ν there), since ν is reserved for indexing individual samples within a batch $\mathcal{D}_t$; and, from Section 4.4 onward, τ denotes continuous time (called t there).

$$\boldsymbol{w}_{j,t+1} = \frac{\boldsymbol{w}_{j,t} - \gamma\nabla_{\boldsymbol{w}_{j,t}}\hat{\mathcal{R}}_t}{\left\|\boldsymbol{w}_{j,t} - \gamma\nabla_{\boldsymbol{w}_{j,t}}\hat{\mathcal{R}}_t\right\|} \qquad \forall t \in [T], \forall j \in [p] \tag{4.4}$$

where:

$$\hat{\mathcal{R}}_t = \frac{1}{2n_b}\sum_{\nu=1}^{n_b}(y^\nu - f(\boldsymbol{z}^\nu))^2, \qquad \forall t \in [T] \tag{4.5}$$

is the empirical risk over a batch of data. Two comments are in place. First, the gradient at each step is computed using the empirical loss given by fresh, previously, unseen samples coming from $\mathcal{D}_t$. Each gradient is thus an unbiased estimator of the true gradient, which means that on average this algorithm minimizes the population risk over W:

$$\mathcal{R} = \mathbb{E}_{(\boldsymbol{z},y)}\left[\frac{1}{2}(y - f(\boldsymbol{z}))^2\right] \tag{4.6}$$

Second, the spherical projection allows us to focus just on the direction learned by the network, putting aside the effect of the change of the norm of the weights. Note that in equation (4.4) we have kept the read-out layer $\boldsymbol{a}$ fixed. Optionally, the second layer could also be trained with SGD, as the first layer, or even with the Moore-Penrose pseudo-inverse solution; In this paper, however, we consider it fixed and focus on the *feature learning* step, i.e., the recovery of the low-dimensional space spanned by $W^\star$.

High-dimensional regime — We focus on the high-dimensional regime where $d \to \infty$. Of particular interest is the case where the batch size n_b scales with the dimension d. Indeed, in modern machine learning, and in particular in the realm of distributed and federated learning, scenarios with large batches, a single pass, and few iterations often become the norm [Goy+17; Li+20] (as for instance when training large language models), further underlining the relevance of this scenario.

More precisely, we assume a scaling of the relevant parameters, i.e., learning rate and the batch size, with d, as follows

$$\gamma = \gamma_0 d^{-\delta} \quad \text{and} \quad n_b = n_0 d^{\mu}, \tag{4.7}$$

with $\mu \geq 0$ and δ could possibly be any real value. The exponents (δ, μ) characterize the Time / Complexity tradeoff illustrated in the phase diagram (Figure 4.1). More precisely, the figure shows the time complexity $T = T_0 d^{\theta}$ as a function of the batch size exponent μ. The time exponent (θ) is linked to the learning-rate exponent (δ) and the information exponent (ℓ), and determining this relation is the main object of analysis of the following sections.

Weak recovery of the target — The central object of our analysis is to characterize the time iterations needed for the SGD dynamics defined in Equation (4.5) to learn the low-dimensional features $W^\star$. More precisely, we are interested in studying the number of steps to achieve order one correlation with the target weights $W^\star$. We refer to this condition as *weak recovery* of the target subspace, formalized in the following definition.

Definition 4.2 (Weak recovery)**.** *The target subspace $V^\star$ is defined as the span of the rows of the target weights $W^\star$:*

$$V^\star = \text{span}(\boldsymbol{w}_1^\star, \dots, \boldsymbol{w}_k^\star) \tag{4.8}$$

We define the following weak recovery stopping time for a parameter $\eta \in (0, 1)$ independent from d:

$$t_\eta^+ = \min\{t \geq 0 : \|W W^{\star\top}\|_F \geq \eta\} \tag{4.9}$$

Our key objective is to characterize the largest affordable batch size n_b to achieve weak recovery of the relevant target subspace $V^\star$ while minimizing the training time iterations T. Indeed, the updates of one-pass SGD in Equation (4.5) consist of sums of independent terms that can be parallelized efficiently with decentralized learning protocols.

Our **main contributions** in this paper are the following:

- We study how the batch size influences the number of steps required to learn a target function, for different information exponents of the problem. We introduce a schematic phase diagram describing the different learning regimes, see Figure 4.1.
- We show that performing gradient updates with large batch sizes can reduce the training time without changing the total sample complexity to weakly recover the teacher subspace only up to $n_b \lesssim d^{\ell/2}$ samples per step, with d the data dimension and ℓ the information exponent of the target. Beyond this limit, larger batch sizes are detrimental for one-pass SGD.
- We characterize that it is possible to improve over this fundamental limitation of one-pass SGD by using gradient updates on the correlation loss, namely *Correlation loss SGD*. We provably show that the number of steps needed to weakly correlate with the target with this new training protocol can then be pushed down to $T = \text{polylog}(d)$ when using batch sizes $n_b = O(d^{\ell-1})$, with ℓ the information exponent. Additionally, we provide sharp prescription on how to scale the learning rate with batch size and input dimension, in order to achieve the best time-memory tradeoff.
- We show that the asymptotic training dynamics is described by a system of Ordinary Differential Equations (ODEs) that can be solved exactly. We leverage on the ODE description to characterize the different learning phases of two-layer networks when initialized with non-vanishing initial correlation with the target direction to be learned (warm starts). Furthermore, we also discuss finite d corrections to the asymptotic dimension-free description.
- Finally, we validate and illustrate our theoretical results with numerical experiments.

The code to reproduce representative figures is available in our GitHub repository[1]. We refer to Appendix D.5 for details on the numerical implementations while the rigorous proofs of the main results are detailed in Appendix D.1.

Other related works — The dynamics of Stochastic Gradient Descent (SGD) in two-layer neural networks, particularly when trained on synthetic Gaussian data, have been a topic of interest since the seminal works in the mid-1990s [SS95b; SS96; BS95; RB95]. This area has experienced a resurgence in recent years [TV23; Gol+19; Vei+22; Arn+23; Arn+24a; BMZ24; BGJ21; Paq+22; Col+24; MBB24].

Many theoretical efforts highlighted the class of functions that are efficiently learned by two layer neural networks. In the context of single-index targets, [BGJ21] introduces the notion of information exponent to quantify the hardness of the learning task. Similarly, for multi-index models, [AAM22; AAM23], building on their earlier work [Abb+21], demonstrated how the leap complexity of target functions dictates the amount of training samples needed from two-layer networks in the mean-field limit to learn the target. Note that [AAM23] also considered the case of $n_b \lesssim O(d)$. Numerous theoretical studies devoted to the understanding of the feature learning regime in two-layer networks often assume an asymptotically vanishing initialization for the second layer weights $\boldsymbol{a}_0$ in Equation (4.1), see e.g. [AAM22; BMZ24; AAM23]. Although this

[1] `https://github.com/IdePHICS/batch-size-time-complexity-tradeoffs`

assumption is amenable for theoretical characterizations, our analysis provably shows that a careful reasoning on the second layer magnitude is needed to offer a complete portrait of the learning dynamics of SGD. More precisely, we describe a sharp divergence when the batch size $n_b \gg d^{\ell/2}$ between the dynamics of SGD when optimizing the MSE loss (vanilla SGD), in contrast to the correlation loss $\tilde{\ell} = \frac{1}{n_b}\sum_{\nu\in[n_b]} 1 - y^\nu f(\boldsymbol{z}^\nu)$ (Correlation loss SGD). The latter training protocol is indeed equivalent to consider an asymptotically vanishing second layer weights $\boldsymbol{a}_0$ at initialization in the optimization routine, e.g. see [Dam+23].

Closer to us, the analysis of the *first* gradient descent step with large n_b has been discussed in detail in recent papers [Ba+22; DLS22; Dan+24a]. [Ba+22] showed that a single large learning rate gradient step allows beating kernel methods when the number of training samples is proportional to the input dimension. While their results are limited to single-index target and to a single gradient step, [DLS22] further showed that with $n = \omega(d^2)$ samples, two-layer nets can learn multi-index target function with zero first Hermite coefficient (ℓ=2). [Dan+24a] extended their conditions on the sample complexity to general $\ell \geq 1$, showed this complexity is optimal for single-step learning, and extended the results to higher information exponents. Although motivated from different objectives, [SW24] heuristically sketch a phase diagram for the performance of SGD on realistic datasets as a function of the algorithm's relevant parameters, i.e. batch size and learning rate.

A common assumption in theoretical studies is to consider sample-splitting schemes for the training protocol. At each iteration, the optimization algorithm is run using a fresh batch of observations of the model, drawn independently of past iterations; this routine has been used extensively in the analysis of iterative algorithms (see e.g. [CPT21; JNS13; HW14; JN15; Kwo+19]).

4.3 Time / Complexity tradeoffs

In this section, we characterize the intertwined dependence between the batch size and the hardness of the learning task in determining the number of one-pass SGD iterations needed to achieve weak recovery of the teacher subspace as in Definition 4.2. We offer a detailed picture of the tradeoffs to consider in order to minimize the training iteration time T, compactly illustrated in the phase diagram in Figure 4.1.

Network initialization — We consider random initialization for the hidden layer weights of the network (4.1), while the second layer weights are kept fixed:

$$\boldsymbol{w}_{j,0} \sim \text{Unif}(\mathbb{S}^{d-1}), \quad a_{j,0} = 1 \qquad j \in [p]. \tag{4.10}$$

We will refer to this situation as *cold start*, since the initial network correlation with the target directions is vanishing when $d \to +\infty$.

Generalized Linear Models — The seminal work of [BGJ21] studies the weak recovery problem for Generalized Linear Models (GLMs), i.e. $p = 1$, when learning single-index targets ($k = 1$). Starting from randomly initialized networks as defined in (4.1), the time iterations needed

for one-pass SGD (with one sample per batch) to achieve weak recovery of the target direction respects:

$$I(\ell) = \begin{cases} O(d^{\ell-1}) & \text{if } \ell > 2 \\ O(d \log d) & \text{if } \ell = 2 \\ O(d) & \text{if } \ell = 1 \end{cases} \tag{4.11}$$

where ℓ is the information exponent of the target $f^\star$.

As far as weak recovery of the target subspace is concerned, the characterization of multi-index targets follows the same lines of thought just replacing the information exponent by the leap index of the target, e.g. see Definition **3** of [Dan+24a] or Definition **1** of [AAM23]. Similarly to the information exponent definition, the leap index is the lowest rank of the tensors appearing in the Hermite expansion of the target $f^\star$. Therefore, we choose to study in the following the training dynamics for the $p = k = 1$ scenario for general batch sizes n_b. This assumption is useful to provide rigorous guarantees as it largely reduces the complexity of the projected SGD dynamics. However, we argue (supported by numerical simulations in Appendix D.5.1) that the same phenomenology will hold for larger values of p and k.

4.3.1 Weak recovery with one-pass SGD

Consider the gradient descent dynamics defined on the hidden layer weights by Equation (4.4). We focus on the description of the time evolution of the correlation between the network's hidden layer weight and the target direction:

$$m_t = \langle \boldsymbol{w}_t, \boldsymbol{w}^\star \rangle \tag{4.12}$$

Our first main result is to characterize the time to achieve weak recovery of the target direction $\boldsymbol{w}^\star$ as a function of the batch size and the information exponent of the target. We make very weak assumptions on the activation and labeling functions, namely only assuming a sub-polynomial growth:

Assumption 4.1 (Polynomial growth)**.** *The activation function σ is differentiable everywhere, except maybe at a finite set of points. Both σ' and $f^\star$ are sub-polynomial, i.e. there exists a $q > 0$ and a constant C such that for any $x \in \mathbb{R}$*

$$|\sigma'(x)| \leq C(1+x)^q \quad \textit{and} \quad |f^\star(x)| \leq C(1+x)^q \tag{4.13}$$

Assumption 4.2 (Well-posedness)**.** *Let $(c_k)_{k\geq 0}$ and $(c_k^\star)_{k\geq 0}$ be the Hermite coefficients of σ and $h^\star$, respectively. Then $c_\ell \neq 0$, and if ℓ is even, then $c_\ell c_\ell^\star > 0$.*

Assumption 4.3 (Initialization)**.** *There exists a $\kappa > 0$ such that $m_0 > \kappa/\sqrt{d}$. Further, if ℓ is odd, then m_0 is such that*

$$c_\ell c_\ell^\star m_0 > 0$$

Assumption 4.2 ensures that the optimization problem is achievable for gradient flow on the population loss $\mathcal{R}$. Indeed, one can show that when $m \approx 0$,

$$\mathcal{R} = R_0 - \ell!\, c_\ell c_\ell^\star m^\ell + o(m^\ell);$$

as a result, if ℓ is even and $c_\ell c_\ell^\star < 0$, then $m = 0$ is a local maximum of $\mathcal{R}$ and weak recovery is impossible. When ℓ is odd, the point $m = 0$ is always a strict saddle, so Assumption 4.3 ensures that we start on the correct side of the saddle. Under the initialization scheme described by Equation 4.10, the first condition is satisfied with arbitrarily high probability upon decreasing κ, while the second is a $1/2$-probability event.

We are now in the position to formally state the result:

Theorem 4.1 (Projected SGD weak recovery)**.** *Consider the projected SGD algorithm with square loss (Equations* (4.4)*,* (4.5)*), and suppose that Assumptions 4.1-4.3 hold. There exist absolute constants c_γ, C_γ such that if*

$$\gamma \le c_\gamma \min\left(1, n_b d^{-\left(\frac{\ell}{2} \vee 1\right)} \log(d)^{-C_\gamma}\right),$$

then for large enough d we have with probability $1 - ce^{-c\log(n)^2}$

$$t_\eta^+ \le C\gamma^{-1} d^{\left(\frac{\ell}{2}-1\right)\vee 0} \log(d). \tag{4.14}$$

4.3.2 Illustration of Theorem 4.1

The phase diagram in Figure 4.1 exemplifies Theorem 4.1. We identify three *learning phases*: SGD learning, Correlation Loss SGD learning, and SGD impossible. These regions are explored by varying the batch size and learning rate exponents δ, μ. Our theory characterizes the optimal learning rate to achieve the lowest possible time iterations of SGD to weakly recover the target direction $\boldsymbol{w}^\star$ when the batch size respects $n_b = o(d^{\ell-1})$:

$$\delta^\star(\mu) = \begin{cases} \frac{\ell}{2} - \mu & \text{if } \mu < \ell/2 \\ 0 & \text{otherwise} \end{cases} \tag{4.15}$$

Weak recovery region — In the region $n_b \lesssim d^{\ell/2}$ there is a net benefit in using larger batch sizes in the SGD optimization. This section shows a similar phenomenology to [BGJ21]: if we optimally choose the learning rate exponent $\delta^\star(\mu)$, the number of time iterations needed to weakly recover the teacher direction $\boldsymbol{w}^\star$ is simply $T(n_b) = I(\ell)/n_b$, rescaling straightforwardly the time complexity of the $n_b = 1$ case in Equation (4.11). By considering higher values for the learning rate ($\delta < \delta^\star(\mu)$) SGD is not able to weakly recover the signal as the dynamics is dominated by terms contracting the network / target correlation to zero, defining the SGD impossible region. Vice versa, if one takes into account lower learning rates ($\delta > \delta^\star(\mu)$), it is certainly possible to weakly-recover the target, but at a higher time complexity cost.

Self-interaction regime — Conversely, the region $d^{\ell/2} \ll n_b \lesssim d^{\max(\ell-1,1)}$ does not adhere to the same straightforward paradigm. Indeed, standard SGD is not able to achieve weak recovery of

the teacher direction using $T(n_b) = {}^{I(\ell)}/_{n_b}$ time iterations, but a simple modification of it - that we call *Correlation Loss SGD* - is able to. The number of steps needed to weakly recover the target with this new training protocol can then be pushed down to $T = \text{polylog}(d)$ when using batch sizes $n_b = O(d^{\max(\ell-1,1)})$. We refer to the next section for a detailed analysis of this regime.

One-step regime — Recent works have discussed the role of one large learning rate gradient descent step (*giant step*) when training of two-layer networks [Ba+22; DLS22; Dan+24a]. More precisely, [Dan+24a] sharply characterizes the section $n_b = \Omega(d^\ell)$ where it is possible to learn the teacher direction in just one step by setting the learning rate to $\delta_{\text{giant}-\text{step}}(\mu) = \frac{1-\ell}{2}$.

4.3.3 The self-interaction regime

Surprisingly, when the learning rate becomes extensive ($\gamma = \omega(1)$), the usual SGD algorithm struggles to achieve weak recovery. This can be explained by writing the gradient update as

$$\boldsymbol{w}_t - \gamma \boldsymbol{g}_t = (1 - \gamma \langle \boldsymbol{g}_t, \boldsymbol{w}_t \rangle) \boldsymbol{w}_t - \gamma \boldsymbol{g}_t^\perp,$$

where $\boldsymbol{g}_t, \boldsymbol{g}_t^\perp$ are the gradient at time t and its component orthogonal to $\boldsymbol{w}_t$, respectively. As a result, projected gradient descent can be seen as a version of spherical SGD with a random weight decay $\gamma \langle \boldsymbol{g}_t, \boldsymbol{w}_t \rangle$. When $\gamma = \omega(1)$, this weight decay also becomes of order $\omega(1)$, which leads to very unpredictable behavior of the process $(\boldsymbol{w}_t)_{t \geq 0}$.

In this section, we study a modified version for the training protocol, in which the self-interaction term $\langle \boldsymbol{g}_t, \boldsymbol{w}_t \rangle$ is much smaller; we will refer to this new algorithm as *Correlation loss SGD* (see e.g. [Dam+23]), as it effectively amounts to gradient updates on the correlation loss:

$$\tilde{\ell} = \frac{1}{n_b} \sum_{\nu \in [n_b]} 1 - y^\nu f(\boldsymbol{z}^\nu) \tag{4.16}$$

The above-described protocol is equivalent to consider a vanishing initialization scale for the second layer weights $\boldsymbol{a}_0$ of the network 4.1. Such assumptions are often considered in different theoretical efforts (see e.g. [AAM22; BMZ24; AAM23]). However, Figure 4.1 illustrates that a careful analysis of the initialization scale $\boldsymbol{a}_0$ is needed to paint an exhaustive description of the SGD dynamics. Indeed, considering Correlation loss SGD allows to overcome the limitations highlighted by Theorem 4.1 for projected SGD. In particular, Correlation loss SGD is able to access the yellow region depicted in Figure 4.1 where the time complexity can be reduced again to $\tilde{T}(n_b) = {}^{I(\ell)}/_{n_b}$ using the optimal learning rate $\tilde{\delta}^\star(\mu) = {}^\ell/_2 - \mu$ even for $\mu > {}^\ell/_2$. This is precisely stated in the following theorem.

Theorem 4.2 (*Correlation Loss SGD* weak recovery)**.** *Consider the projected SGD algorithm with correlation loss (Equations* (4.4)*,* (4.16)*), and suppose that Assumptions 4.1-4.3 hold. There exists absolute constants c_γ, C_γ such that if*

$$\gamma \leq c_\gamma \log(d)^{-C_\gamma} \min\left(n_b d^{-\left(\frac{\ell}{2} \vee 1\right)}, \sqrt{\frac{n_b}{d}} \right)$$

ℓ	SGD with $n_b \lesssim d^{\ell/2}$	SGD with $d^{\ell/2} \ll n_b \lesssim d^{\max(\ell-1,1)}$	Correlation loss SGD with $n_b = o\left(d^{\max(\ell-1,1)}\right)$	One step with $n_b = O(d^\ell)$
1	$T = O(d/n_b)$	$T = O(1)$	$T = O(d/n_b)$	$T = 1$
	$N = O(d)$	$N = O(d)$	$N = O(d)$	$N = O(d)$
2	$T = O(d \log d/n_b)$	$T = O(\log d)$	$T = O(d \log d/n_b)$	$T = 1$
	$N = O(d \log d)$	$N = O(d \log d)$	$N = O(d \log d)$	$N = O(d^2)$
> 2	$T = O(d^{\ell-1}/n_b)$	$T = O(d^{\ell/2-1})$	$T = O(d^{\ell-1}/n_b)$	$T = 1$
	$N = O(d^{\ell-1})$	$N = O(n_b d^{\ell/2-1})$	$N = O(d^{\ell-1})$	$N = O(d^\ell)$

Table 4.1: **Time / Complexity tradeoffs:** Number of iterations T and the total number of samples N needed to achieve weak recovery of the target for different training protocols in high dimensions. **Left:** One-pass SGD of batch size $n_b = d^{\ell/2}$, in this regime the optimal time complexity is obtained rescaling by n_b the result of [BGJ21] for $n_b = 1$, i.e. by choosing the optimal learning rate $\gamma = O(n_b d^{-\ell/2})$. **Center-left:** One-pass SGD with batch size $d^{\ell/2} \ll n_b \lesssim d^{\max(\ell-1,1)}$, for hard problems ($\ell > 2$) the sample complexity is significantly increased with respect to the $n_b = 1$ case up to $N = O(n_b d^{\ell/2-1})$. The learning rate cannot be increased proportionally to n_b in this region, fixed to be $\gamma = O(1)$. **Center-Right:** Correlation loss SGD with $n_b = o(d^{\max(\ell-1,1)})$, this training protocol overcomes the limitation of SGD when $n_b \gg d^{\ell/2}$ and $\ell > 2$; the total sample complexity is $N = O(d^{\ell-1})$. The learning rate is fixed again to be proportional to the batch size $\gamma = O(n_b d^{-\ell/2})$. **Right:** The target is weakly recovered with one GD step of $n_b = O(d^\ell)$ batch. The learning rate is chosen as $\gamma = O(d^{(\ell-1)/2})$ for the One Step routine [Dan+24a].

Then if d is large enough, we have with probability $1 - ce^{-c\log(n)^2}$

$$t_\eta^+ \le C\max\left(1, \gamma^{-1} d^{\left(\frac{\ell}{2}-1\right)\vee 0} \log(d)\right). \tag{4.17}$$

The derivation of Theorems 4.1 and 4.2 generalizes [BGJ21] which studies the $n_b = 1$ case. Informally, the result is obtained by analyzing the stability of the equation for the correlation m_t, along with the requirement on the step-size for the suppression of the effects of the noise across time. However, there is a major difficulty introduced by the large step-size regime: when the gradient updates become larger, the Taylor-inspired bounds used in [BGJ21] become vacuous. We work around this problem by showing that in this regime, there is a *one-step improvement* which jumps directly to meaningful correlation with the target vector. All details can be found in Appendix D.1. We provide in Table 4.1 a representative summary of the results in Theorems (4.1. 4.2) characterizing the time/complexity tradeoffs to achieve weak recovery of general single index target $f^\star$.

The theoretical predictions of Theorem 4.2 are evaluated in Figure 4.2. The plot compares the student-teacher weight correlation ($m_t = \langle \boldsymbol{w}_t, \boldsymbol{w}^\star \rangle$) achieved by vanilla projected SGD and *Cor-*

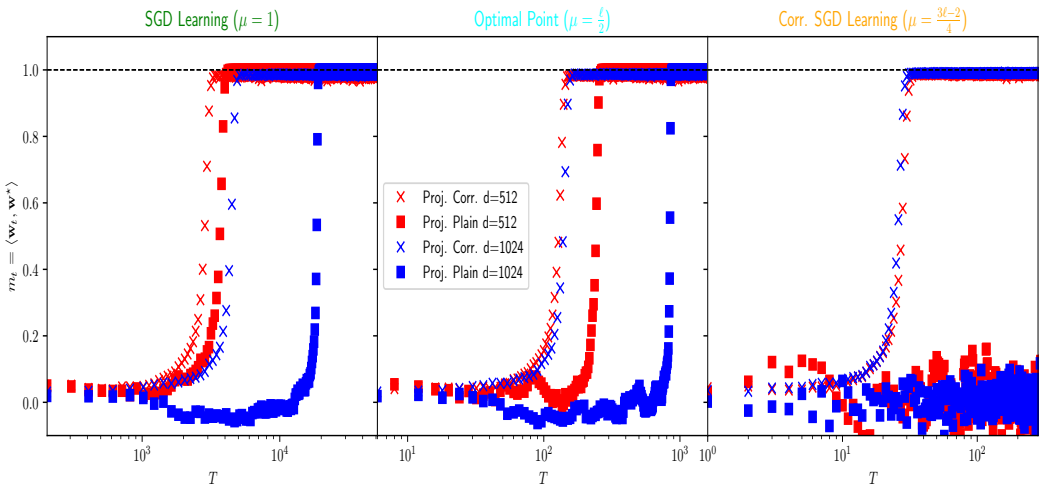


Figure 4.2: **Correlation Loss SGD weak recovery:** Comparison between the performance of plain SGD and the Correlation Loss SGD, in different regions of the phase diagram, and for different sizes d. The plot shows the test error as a function of the optimization steps. Both the teacher and the student activation functions are fixed to $\sigma = h^\star = \mathrm{He}_3$, so the information exponent is $\ell = 3$. In all the three plots we vary the value of μ, while $\delta = \ell/2 - \mu$. Theorem 4.2 predicts that the *Correlation Loss SGD* weakly recovers the target direction while SGD fails when $\delta < 0$, in accordance to what is shown in the plot. Note that the numbers of steps needed for the target recovery drastically decrease when μ becomes large in accordance with Theorems 4.1,4.2.

relation Loss SGD as a function of time. The teacher activation $h^\star$ is fixed to be the third Hermite polynomial ($\ell = 3$), and the batch size varies, effectively changing the region of the phase diagram considered. In agreement with Theorem 4.2, the figure shows that *Correlation Loss SGD* is always able to achieve faster weak recovery with respect to SGD. Furthermore, the batch size that can be used with *Correlation Loss SGD* in combination with the optimal learning rate $\tilde{\delta}^\star(\mu) = \ell/2 - \mu$ is larger, as presented in the phase diagram of Figure 4.1.

Remark 4.1. *Theorem 4.2 does not claim superiority of* Correlation Loss SGD *with respect to plain SGD when trying to fully learn the target, but only for achieving weak-correlation faster (Definition 4.2). As Figure 4.2 shows,* Correlation Loss SGD *escapes the initial dynamical plateau faster, but is then limited by a loss function not designed properly to reach the global minimum. In Appendix D.5 we investigate the possibility to combine both the algorithms sketched in Figure 4.2, namely escaping the initialization plateau with Correlation loss SGD and then learn the function with SGD; we refer to this protocol as Adaptive SGD. Moreover, as Figure 4.1 and Table 4.1 illustrate, the benefits of using Correlation loss SGD are limited to settings in which $\ell > 2$. Indeed, the Self-interaction regime (depicted in yellow in Figure 4.1) is not present for $\ell \leq 2$.*

4.4 Exact asymptotic description

We now characterize the exact asymptotic description of the dynamics of two-layer networks trained with SGD. In Figure 4.3 we sketch a representative phase diagram as a function of the relevant parameter of the algorithm, i.e. the learning rate and the batch size. The plot identifies different regions of parameters defining the network's learning efficiency.

Sufficient statistics — Our study, like many other efforts [BGJ22; SS95b], is based on the concentration of the neurons' overlaps with the target subspace and their norms. This approach only requires the knowledge for every optimization step $t \in [T]$ of the above defined overlaps, often referred to as *sufficient statistics*. Let the pre-activations be defined as:

$$\boldsymbol{\lambda}_t = W_t \boldsymbol{z} \qquad \text{and} \qquad \boldsymbol{\lambda}^\star = W^\star \boldsymbol{z} \tag{4.18}$$

Thanks to the Gaussian nature of the data, the pre-activations at any time step t are jointly Gaussian vectors $(\boldsymbol{\lambda}_t, \boldsymbol{\lambda}^\star) \sim \mathcal{N}(\mathbf{0}_{p+k}, \Omega_t)$ with covariance $\Omega_t \in \mathbb{R}^{(p+k)\times(p+k)}$:

$$\Omega_t \coloneqq \begin{pmatrix} Q_t & M_t \\ M_t^\top & P \end{pmatrix} = \begin{pmatrix} W_t W_t^\top & W_t W^{\star\top} \\ W^\star W_t^\top & W^\star W^{\star\top} \end{pmatrix} \tag{4.19}$$

We refer to M_t, Q_t as the *order parameters*.

4.4.1 Closed form equations

We are now in the position to state our proposition that provides a set of deterministic ODEs to describe one-pass SGD in high-dimensions. This portrayal depends ultimately only on the values of the learning rate and the batch size, as quantified by the exponents (δ, μ).

Proposition 4.1. *Consider $\bar{\Omega}(t)$ the solution of the system of ordinary differential equations*

$$\begin{aligned} \frac{\mathrm{d}M_{jr}}{\mathrm{d}\tau} =& \Psi_{jr}(\Omega) - \frac{M_{jr}}{2}\Phi_{jj}(\Omega) \\ \frac{\mathrm{d}Q_{jl}}{\mathrm{d}\tau} =& \Phi_{jl}(\Omega) - \frac{Q_{jl}}{2}\left(\Phi_{jj}(\Omega) + \Phi_{ll}(\Omega)\right) \end{aligned} \tag{4.20}$$

where we introduced:

$$\begin{aligned} \Psi_{jr}(\Omega) =& \mathbf{1}_{\{\delta\geq 0 \bigcap 2\delta+\mu\geq 1\}} \frac{\gamma_0}{p} a_j \psi_{jr} \\ \Phi_{jl}(\Omega) =& \mathbf{1}_{\{\delta\geq 0 \bigcap 2\delta+\mu\geq 1\}} \frac{\gamma_0}{p}\left(a_j \phi_{jl}^{\mathrm{GF}} + a_l \phi_{lj}^{\mathrm{GF}}\right) \\ &+ \mathbf{1}_{\{\delta+\mu\geq 1 \bigcap 2\delta+\mu\leq 1\}} \frac{\gamma_0^2}{p^2 n_0} a_j a_l \phi_{jl}^{\mathrm{HD}} \end{aligned} \tag{4.21}$$

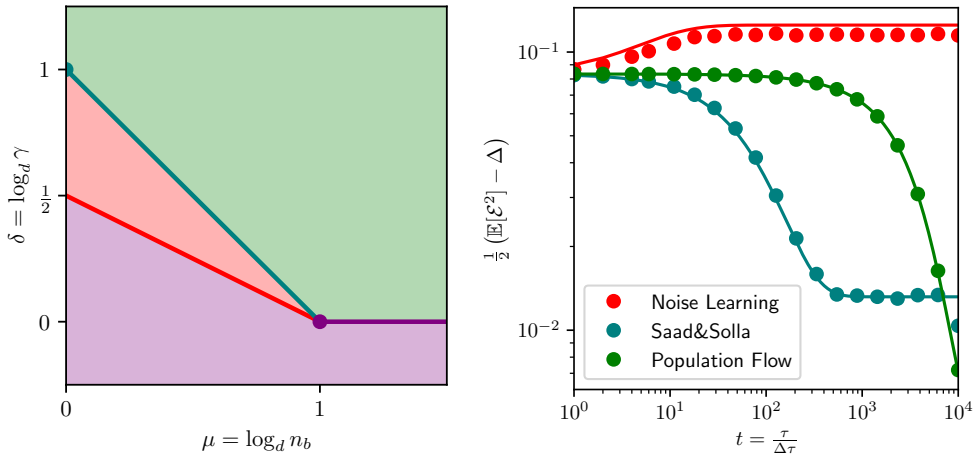


Figure 4.3: **Exact asymptotic description:** Exact asymptotic characterization of the dynamics of two-layer networks trained with SGD as a function of the batch size (n_b) and the learning rate (γ). **Left:** Illustration of the different dynamical regimes in a compact phase diagram. **Population flow region:** The dynamics is equivalent to population gradient flow. **Noise learning region:** The high-dimensional noise terms dominate the dynamics. **Saad & Solla line:** The learning dynamics attains a plateau characterized by the noise variance [SS95b]. **Dynamics not defined:** The deterministic low-dimensional description of the Equation (4.20) is not valid. **Right:** The figure shows a comparison of numerical simulations (dots) and theoretical prediction (continuous lines) for three instances (δ, μ) associated with different learning regimes (identified by the corresponding colors). For both theory and simulations, the test error is plotted as a function of SGD iterations. We consider a matching architecture problem, i.e. $h^\star = \sigma =$ erf activation, and hidden units $p = 2, k = 2$.

and auxiliary integrals bearing expectations over $\mathcal{N}(\mathbf{0}, \Omega)$*:*

$$\begin{aligned}\psi_{jr} =& \mathbb{E}\left[\sigma'(\lambda_j)\lambda_r^\star \mathcal{E}\right]\\ \phi_{jl}^{\mathrm{GF}} =& \mathbb{E}\left[\sigma'(\lambda_j)\lambda_l \mathcal{E}\right]\\ \phi_{jl}^{\mathrm{HD}} =& \mathbb{E}\left[\sigma'(\lambda_j)\sigma'(\lambda_l)\mathcal{E}^2\right]\\ \mathcal{E} =& g_\star(\boldsymbol{\lambda}^\star) - \frac{1}{p}\sum_{j=1}^{p} a_j \sigma(\boldsymbol{\lambda})\end{aligned}$$

Then, there exists a constant C *independent from the input data dimension, such that the discrete stochastic process for the covariance* $\{\Omega_t\}_{t\in\mathbb{N}}$ *in Equation* (4.19) *induced by projected SGD dynamics is approximated by the deterministic covariance matrix* $\bar{\Omega}(t)$ *with precision:*

$$\mathbb{E}\left\|\Omega_t - \bar{\Omega}(t\Delta\tau)\right\| \leq e^{C\tau}\sqrt{\Delta\tau} \tag{4.22}$$

with $\tau = t\Delta\tau$ *and* $\Delta\tau = d^{\max(-\delta,-2\delta+1-\mu)}$.

We refer to Appendix D.2 for the informal derivation of the above result.

In Figure 4.3 (left) we summarize the results of Proposition 4.1 in a compact phase diagram. The following dynamical regimes appear:

- **Population Flow**: The dynamics of the sufficient statistics described by a deterministic set of ODEs (4.20) is equivalent to population gradient flow.
- **Noise learning**: The dynamic is dominated by high-dimensional noise, and consequently the algorithm does not learn the target; the behavior is reflected in the ODEs.
- **Saad & Solla line**: The ODE description in Proposition 4.1 is equivalent to the pivotal work on 2LNNs [SS95b]. In particular, the original work corresponds to the point $(\delta, \mu) = (1, 0)$. The learning dynamics is blocked on a plateau characterized by the noise variance in the labels.
- **Large batch interaction**: The dynamics is given by population gradient flow combined with the large-batch corrections terms, i.e. contributions coming from interaction between samples $\boldsymbol{z}^\nu, \boldsymbol{z}^{\nu'}$ in the same batch. This region corresponds in parameter space to $\delta = 0, \mu > 1$.
- **Phase coexistent point**: The point $\delta = 0, \mu = 1$ is the only point in parameter space where all term in the equations are not negligible.
- **Dynamics not defined**: For a broad range of values of (δ, μ) the SGD dynamics is not effectively described by a set of low-dimensional deterministic ODEs.

In the right panel of Figure 4.3 we present a numerical investigation of three particular instances of the regimes presented above. The plot shows a comparison of numerical simulations versus the low-dimensional exact asymptotic characterization given in Equations (4.20). The values of the learning rate and the batch size used for SGD training are varied to probe different regions of phase diagram 4.3.

Remark 4.2. *When the target's leap index is $\ell > 1$, the dynamic of SGD is dominated by a first extensive search phase to achieve weak recovery of the teacher direction (Theorem 4.1). Therefore, in order to probe interesting dynamical regimes for general single index teachers, we assume non-vanishing initial correlations of the network's hidden layer weights with the teacher's ones when $d \to +\infty$.*

In Appendix D.5 we study the tightness of the exponential bound Equation (4.22)*; we argue supported by numerical illustrations that (on the practical side) the low dimensional ODE description is valid well beyond the extents of Proposition 4.1, as already observed by other works [Gol+19; Vei+22] in different context.*

Non-asymptotic corrections — Proposition 4.1 unveils a surprising result for the exact asymptotic description of two-layer networks. Indeed, the ODEs written in Equation (4.20) coincide with the analogous ones for the single sample per batch case ($n_b = 1$), modulo trivial rescaling of the parameters [Vei+22]. However, a careful consideration of the self-interaction term in the gradient is needed for correctly describing the low-dimensional process of the order parameters:

$$\sum_{\nu'=1,\nu'\neq\nu}^{n_b} \sigma'(\lambda_j^\nu)\sigma'(\lambda_l^{\nu'})\mathcal{E}^\nu\mathcal{E}^{\nu'}\langle \boldsymbol{z}^\nu, \boldsymbol{z}^{\nu'}\rangle \tag{4.23}$$

The asymptotic form of this term can be exactly computed to be (using Proposition 4.1 notations):

$$\phi_{jl}^{\mathrm{BC}} = \mathbb{E}\left[\sigma'(\lambda_j)\mathcal{E}\left(\lambda^\star\right)^\top\right] P^{-1}\mathbb{E}\left[\sigma'(\lambda_l)\mathcal{E}\lambda^\star\right] + \tag{4.24}$$

$$\mathbb{E}\left[\sigma'(\lambda_j)\mathcal{E}(\lambda^\perp)^\top\right]\left(Q^\perp\right)^{-1}\mathbb{E}\left[\sigma'(\lambda_l)\mathcal{E}\lambda^\perp\right] \tag{4.25}$$

with $\boldsymbol{\lambda}^\perp = \boldsymbol{\lambda} - MP^{-1}\boldsymbol{\lambda}^\star$ and $Q^\perp = Q - MP^{-1}M^\top$. Although the contribution of the above term is asymptotically vanishing in the ODE description (4.20) when $d \to \infty$, any theoretical description at finite d will effectively depend on ϕ_{jl}^{BC}. In Appendix D.5 we provide additional numerical investigation on the importance of (4.24) and the role of large batch sizes for non-asymptotic corrections to the characterization in Proposition 4.1. Moreover, we note that taking into account the presence of large batch size is pivotal to illustrate the time / complexity tradeoffs for weak recovery of the target subspace, as thoroughly discussed in Section 4.3.

4.5 Conclusions

In this manuscript, we have explored the intricate relationship between batch size and the efficiency of learning multi-index targets using one-pass SGD on high-dimensional input data. Our findings defy the conventional belief that larger batch sizes invariably lead to better results and reveal a critical batch size threshold, beyond which the advantages of larger batches wane in terms of computational complexity. Applying gradient updates on the correlation loss one may, however, navigate this limitation. Finally, we also provide a system of low-dimensional ODE to describe the exact asymptotic of the SGD dynamics with arbitrary batch-sizes. Moving forward, we hope this research paves the way for deeper inquiries into the optimization behaviors of learning algorithms, prompting further examination of deeper networks and alternative loss functions.

Acknowledgements

This work was supported by the Swiss National Science Foundation under grant SNSF OperaGOST (grant number 200390) and the Choose France - CNRS AI Rising Talents program.

Chapter 5

Repetita Iuvant: Data Repetition Allows SGD to Learn High-Dimensional Multi-Index Functions

Luca Arnaboldi, Yatin Dandi, Florent Krzakala, Luca Pesce, and *Ludovic Stephan*

Abstract. Neural networks can identify low-dimensional relevant structures within high-dimensional noisy data, yet our mathematical understanding of how they do so remains scarce. Here, we investigate the training dynamics of two-layer shallow neural networks trained with gradient-based algorithms, and discuss how they learn pertinent features in multi-index models, that is target functions with low-dimensional relevant directions. In the high-dimensional regime, where the input dimension d diverges, we show that a simple modification of the idealized single-pass gradient descent training scenario, where data can now be repeated or iterated upon twice, drastically improves its computational efficiency. In particular, it surpasses the limitations previously believed to be dictated by the Information and Leap exponents associated with the target function to be learned. Our results highlight the ability of networks to learn relevant structures from data alone without any pre-processing. More precisely, we show that (almost) all directions are learned with at most $O(d \log d)$ steps. Among the exceptions is a set of hard functions that includes sparse parities. In the presence of coupling between directions, however, these can be learned sequentially through a hierarchical mechanism that generalizes the notion of staircase functions. Our results are proven by a rigorous study of the evolution of the relevant statistics for high-dimensional dynamics.

5.1 Introduction

Gradient Descent-based algorithms such as Stochastic Gradient Descent (SGD) and its variations, are a fundamental tool in training neural networks, and are therefore the subject of intense theoretical scrutiny. Recent years have witnessed significant advancements in understanding their dynamics and the learning mechanisms of neural nets. Significant progress has been made, in

particular, in the case of two-layer networks thanks, in part, to the so-called mean-field analysis [MMN18; CB18; RV22; SS20]. A large part of the theoretical approach focused on *one-pass* optimization algorithms, where *each iteration involves a new fresh batch of data*. In particular, in the mathematically solvable case of high-dimensional synthetic Gaussian data, and a low-dimensional multi-index target function model, the class of functions efficiently learned by these *one-pass* methods has been thoroughly analyzed in a series of recent works, and has been shown to be limited by the so-called information exponent [BGJ21] and leap exponent [AAM22; AAM23]. These analyses have sparked many follow-up theoretical works over the last few months, see, e.g. [DLS22; Dam+23; Dan+24a; BBP23; Ba+23; Mon+24; Mou+23; ZB24], leading to a picture where SGD dynamics is governed by the information exponent, that is, the order of the first non-zero Hermite coefficient of the target.

A common point between all these works, however, is that they considered the idealized situation where one processes data one sample at a time, all of them independently identically distributed, without any repetition. This is, however, not a realistic situation in practice. Indeed, most datasets contain similar data-points, so that repetition occurs (see e.g. for repetition and duplicate in CIFAR [BD20]). Additionally, it is common in machine learning to repeatedly go through the same mini-batch of data multiple times over epochs. Finally, many gradient algorithms, often used in machine learning in Lookahead optimizers [Zha+19], Extragradient methods [Kor76], or Sharpness Aware Minimization (SAM) [For+21], *explicitly* take a few gradient steps over the exact same data-point. It is thus natural to wonder if using such extra-gradient algorithms, or simply reusing many times some data, would change the picture with respect to the current consensus.

There are good reasons to believe this to be the case. Indeed, it has been observed very recently [Dan+24b] that iterating twice over large ($O(d)$) datasets was enough to alter the picture reached in [BGJ21; AAM22; AAM23]. Importantly, [Dan+24b] showed that there exist functions that can be learned over few (just two) batch repetitions, while they require a much larger (i.e. polynomial in d) number of steps otherwise. While the set of function considered in this work was limited, the conclusion is indeed surprising, thus motivating the following question:

Can data repetition increase the efficiency of SGD when learning any multi-index functions?

We establish a positive answer to the above question by the analysis of more realistic optimization schemes with respect to the standard One-Pass SGD. Indeed, the hardness exponents developed in the seminal works [Abb+21; BGJ21] are substantially bound to the idealized training scenario that considers i.i.d data. Slight modifications to this training scenario toward a more realistic setting starkly change the global picture. Such slight modifications include:

(i) non-vanishing correlations between different data points, that is a natural requirement for real datasets;

(ii) processing the same data point multiple times in the optimization routine, i.e., a standard step in any algorithmic procedure when looping over different epochs.

We consider a simple yet analytically tractable single-pass training scheme: processing one Gaussian sample at a time, but *repeating the gradient step twice*. This minor adjustment, common in Lookahead optimizers [Zha+19], Extragradient methods [Kor76], and Sharpness-Aware Mini-

mization (SAM) [For+21], significantly boosts efficiency. In fact, we will see that this approach matches the performance of our best algorithms in most cases.

We show in particular that most multi-index functions are learned with total sample complexity either $O(d)$ or $O(d \log d)$ if they are related to the presence of a symmetry, without any need for preprocessing of the data that is performed in various works (See e.g., [MM18; LAL19; Mai+20; CM20]). Our results thus demonstrate that shallow neural networks turn out to be significantly more powerful than previously believed [BGJ21; AAM22; AAM23; Arn+23] in identifying relevant features in the data. In fact, they are shown in many (but not all) cases to saturate the bounds predicted by statistical queries (SQ) [Dam+24] (rather than the more limited correlation statistical queries (CSQ) bounds that were characteristic of the former consensus [BGJ21; AAM23]). Equivalently, it roughly mirrors the prediction of the approximate message passing analysis in [Tro+25a]. Our conclusions follow from the rigorous analysis of the overlaps between the trained weight vectors with the low-dimensional relevant subspace that draws inspiration from the pivotal works of [SS95b; CS00; CSX00; BGJ21].

Concurrently with the present work, [Lee+24] also studies the properties of SGD under data repetition in neural networks. We discuss the differences between our works in Section 5.3.2.

5.2 Setting and Main Contributions

Given a dataset of N labeled examples $\{\boldsymbol{z}^\nu, y^\nu\}_{\nu\in[N]}$ in d dimensions, we analyze the learning properties of fully connected two-layer networks with first layer weights $W = \{\boldsymbol{w}_j\}_{j\in[p]}$, second layer $\boldsymbol{a} \in \mathbb{R}^p$, and activation function $\sigma : \mathbb{R} \to \mathbb{R}$

In this chapter the input covariate is denoted $\boldsymbol{z}$, playing the role of $\boldsymbol{x}$ in the rest of the thesis.

$$f(\boldsymbol{z}; W, \boldsymbol{a}) = \frac{1}{p}\sum_{j=1}^{p} a_j \sigma\left(\langle \boldsymbol{z}, \boldsymbol{w}_j\rangle\right) \tag{5.1}$$

We consider target functions that are dependent only on few orthogonal relevant directions $k = O_d(1)$ encoded in the target's weight matrix $W^\star = \{\boldsymbol{w}_r^\star\}_{r\in[k]} \in \mathbb{R}^{k\times d}$:

The target $f^\star$ plays the role of the teacher and the network f that of the student from Section 1.2; we switch to *target/network* here because $f^\star$ and f no longer share the same architecture.

$$y^\nu = f^\star(\boldsymbol{z}^\nu) = h^\star(W^\star \boldsymbol{z}^\nu) \tag{5.2}$$

This model for structured data is usually referred to as a *multi-index* model. Our main objective will be to analyze how efficiently two-layer nets adapt to this low-dimensional structure during training. To ensure that all necessary Gaussian expectations exist, we make the following non-restrictive assumption on $\sigma, h^\star$:

Assumption 5.1. *The functions σ and $h^\star$ have at most polynomial growth, in the sense that there exists a constant $C > 0$ such that for all $x \in \mathbb{R}, \boldsymbol{x} \in \mathbb{R}^k$,*

$$|\sigma(x)| \le C(1+|x|)^C \quad \textit{and} \quad |h^\star(\boldsymbol{x})| \le C(1+\|\boldsymbol{x}\|)^C. \tag{5.3}$$

Training algorithm – Our goal is to minimize the following objective:

$$\mathcal{R}(W, \boldsymbol{a}) = \mathbb{E}_{\boldsymbol{z},y}\left[\mathcal{L}(f(\boldsymbol{z}, W, \boldsymbol{a}), y)\right] \tag{5.4}$$

where $\mathcal{L} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a loss function, and the expectation is over the data distribution.

This objective is non-convex, and we do not have direct access to the expectation over the data. A central question in Machine Learning is therefore to quantify the properties of the weights attained at convergence by different algorithms. In this manuscript we consider a specific class of one-pass training routines to minimize the empirical risk. First, we partition the dataset in disjoint mini-batches of size n_b, i.e. $\mathcal{D} = \cup_{t\in[T]}\{\boldsymbol{z}^\nu, y^\nu\}_{\nu\in\{tn_b+1,\dots,(t+1)n_b\}}$ where T is the total number of iterations considered. For each algorithmic step, we update the hidden layer neurons $W = \{\boldsymbol{w}_j\}_{j\in[p]}$ with the program $\mathcal{A} : (\boldsymbol{w}_{j,t}, \gamma, \rho, n_b) \in \mathbb{R}^{d+3} \to \boldsymbol{w}_{t+1,j} \in \mathbb{R}^d$.

This chapter departs from the discrete/continuous time convention of Section 1.2: here t denotes the discrete SGD step (called ν there), since ν is reserved for indexing the individual samples $\boldsymbol{z}^\nu$.

Algorithm 5.1 (Optimizer Main Step)**.**

$$\begin{aligned}\boldsymbol{w}_{t+1,j} &= \mathcal{A}(\boldsymbol{w}_t, \gamma, \rho, n_b) = \boldsymbol{w}_{t,j} - \frac{\gamma}{n_b}\sum_{\nu\in[n_b]} \nabla_{\boldsymbol{w}_{t,j}}\mathcal{L}\left(f(\boldsymbol{z}^\nu; \tilde{W}_t(\rho), \boldsymbol{a}_0), y^\nu\right)\\ \tilde{\boldsymbol{w}}_{t,j}(\rho) &= \boldsymbol{w}_{t,j} - \rho\nabla_{\boldsymbol{w}_{t,j}}\mathcal{L}\left(f(\boldsymbol{z}^\nu; W_t, \boldsymbol{a}_0), y^\nu\right)\end{aligned} \tag{5.5}$$

This class of algorithms defined by the optimizer step $\mathcal{A}(\cdot, \gamma, \rho, n_b)$, in which we consider the final gradient update after taking a linear combination of the current iterate and its current gradient (noted as $\tilde{W}_t(\rho)$ in the above), is broadly used in different contexts. Indeed, routines with positive ρ parameters correspond to Extragradient methods (EgD) [Kor76], while with negative ρ have been recently used in the context of Sharpness Aware Minimization (SAM) [For+21]. For clarity, we present our theoretical results with the above, easily interpretable, Algorithm 5.1. However, our theoretical claims are valid in a more general setting; we refer to Appendices E.2 and E.4 for additional results on more general optimizer steps.

As previously stated, the central object of our analysis is the efficiency of the network to adapt to the low-dimensional structure identified by $W^\star$. Therefore, we focus on the learned representations by the first layer weights W_t while keeping the second layer weights $\boldsymbol{a}_t = \boldsymbol{a}_0$ fixed during training. This assumption is favorable to performing the theoretical analysis and is largely used in the theoretical community (e.g., [DLS22; Ba+22; Bie+22]).

Numerous theoretical efforts have been devoted to understanding the sample complexity needed to learn multi-index targets. However, an important aspect of many algorithmic routines considered is to exploit a clever *warm start* to initialize the iterates [MM18; LAL19; Mai+20; CM20]. We show in this work that such preprocessing of the data is not needed to learn all polynomial multi-index functions in $T = O(d\log d)$ with the routine in Algorithm 5.1. Instead, we consider a fully agnostic initialization of the network:

Assumption 5.2. *For each $j \in [p]$, the initial first-layer weight $\boldsymbol{w}_{0,j}$ is initialized according to the uniform measure on the sphere:*

$$\boldsymbol{w}_{0,j} \overset{i.i.d}{\sim} \mathrm{Unif}(\mathbb{S}^{d-1}).$$

The second-layer weights are drawn i.i.d in $\{-1, 1\}$:

$$a_{0,j} \overset{i.i.d}{\sim} \mathrm{Unif}(\{-1, 1\}).$$

Weak recovery in high dimensions – The essential object in our analysis is the evolution of the correlation between the hidden neurons $W = \{\boldsymbol{w}_j\}_{j\in[p]}$ and the target's weights $W^\star$, as a function of the algorithmic time steps. More precisely, we are interested in the number of algorithmic steps needed to achieve an order one correlation with the target's weights $W^\star$, known as *weak recovery*.

Definition 5.1 (Weak recovery)**.** *The target subspace $V^\star$ is defined as the span of the rows of the target weights $W^\star$:*

$$V^\star = \text{span}(\boldsymbol{w}_1^\star, \dots, \boldsymbol{w}_k^\star) \tag{5.6}$$

We define the weak recovery stopping time for a parameter $\eta \in (0,1)$ independent from d as:

$$t_\eta^+ = \min\{t \geq 0 : \|WW^{\star\top}\|_F \geq \eta\} \tag{5.7}$$

From Information Exponents to Generative Exponents – Recent works have sharply theoretically characterized the weak recovery stopping time for a variety of multi-index targets learned with one-pass SGD. [BGJ21] for the single-index case, then [AAM22] for the multi-index one have unveiled the presence of an Information Exponent ℓ characterizing the time complexity needed to weakly recover the target subspace $V^\star$, defined in the single-index case as

$$\ell = \min\{k \in \mathbb{N} : \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[h^\star(x)H_k(x)\right] \neq 0\}, \tag{5.8}$$

where H_k is the k-th Hermite polynomial [ODo14]. Under this framework, the sample complexity required to achieve weak recovery when the link function has Information Exponent ℓ is [BGJ21]

$$T(\ell) = \begin{cases} O(d^{\ell-1}) & \text{if } \ell > 2 \\ O(d\log d) & \text{if } \ell = 2 \\ O(d) & \text{if } \ell = 1. \end{cases} \tag{5.9}$$

These bounds have been improved by [Dam+23] up to order $d^{\ell/2}$; the latter matches the so-called *Correlated Statistical Query* lower bound, which considers queries of the form $\mathbb{E}\left[y\phi(\boldsymbol{z})\right]$.

However, this Information Exponent is not the right measure of complexity for low-dimensional weak recovery. Some iterative algorithms, for instance, are known to perform drastically better[Bar+19; CCM21; MM18; LAL19; Mai+20; Tro+25a]. Recently [Dam+24] introduced a new Generative Exponent $\ell^\star$ (generative exponent) governing the weak recovery time scaling for algorithms in the Statistical Query (SQ) or Low Degree Polynomial (LDP) family. This exponent is defined as

$$\ell^\star = \min\{k \in \mathbb{N} : \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[f(h^\star(x))H_k(x)\right] \neq 0 \text{ for some } f : \mathbb{R}\to\mathbb{R}\}, \tag{5.10}$$

which allows for the application of arbitrary transformations of the labels before performing the queries. Under this definition, the "hard" problems with $\ell^\star > 2$ are also impossible to learn for the best first-order algorithm, *Approximate Message Passing* [Tro+25a].

We discuss in this work the extension of the generative exponent to the multi-index setting and investigate the dynamics of specific gradient-based programs (Algorithm 5.1) as a function of the freshly defined hardness exponent $\ell^\star$. These theoretical findings match the recent one of [Kou+24] on the computational efficiency of SGD in learning sparse $k-$parity (associated with $\ell^\star = k$).

When the optimal transformation f is known, a simple SGD algorithm on $(\boldsymbol{z}, f(y))$ achieves the Generative Exponent lower bound. However, one problem remains: does there exist a class of SGD-like algorithms that can achieve this lower bound agnostically?

Achieving lower bounds with data repetition – Although the performance of single-pass SGD is limited by the Information Exponent of $h^\star$, the situation drastically changes when multiple-pass algorithms are considered. Recently, [Dan+24b] proved that non-even single-index targets are weakly recovered in $T = O(d)$ when considering extensive batch sizes with multiple pass. This begs the question of how good can a "reusing" algorithm be. We answer this question for the class of polynomial link functions:

We quote the sharp bound; the additional, not-provably-optimal polylogarithmic factor at generative exponent ≥ 2 is suppressed here and carried in full $(d\log(d)^2)$ by the formal Theorem 5.2.

Theorem 5.1 (Informal)**.** *There is a choice of hyperparameters such that if $h^\star$ is a polynomial function, Algorithm 5.1 achieves weak recovery in $O(d\log d)$ samples.*

This shows in particular that a simple gradient-based algorithm, which only requires seeing the data twice, is optimal—in terms of SGD performance—for weak recovery of any polynomial target.

Learning representations – Learning relevant features from data is a fundamental property of neural networks. However, much theoretical work has focused on lazy training in two-layer networks [JGH18], where features are not effectively learned. In this regime, shallow networks behave as kernel machines, which struggle to adapt to low-dimensional structures in data [Gho+20]. Understanding representation learning and corrections to the kernel regime is thus a key challenge in machine learning theory [DH18; ABP22; Bie+22; DLS22; Pet+22]. In this manuscript we provably characterize the number of iterations needed from Algorithm 5.1 to learn the relevant features of the data, i.e., attain weak recovery of the target subspace. Our results provide the scaling of the relevant hyperparameters (γ, ρ) with the diverging input dimension d. The minibatch size will be fixed to $n_b = 1$ in the main body, and we refer to Appendix E.4 for the extension to larger batch sizes.

Summary of Main Results

- We prove for single-index targets that one-pass gradient-based algorithms surpass the Correlation Statistical Query (CSQ) limits established by the Information Exponent [BGJ21].
- We unveil that the dynamics of the family of Algorithm 5.1 are governed by the generative exponent [Dam+24], with an additional polynomial restriction on the transformation of the output.
- We generalize the notion of generative exponent to multi-index targets. As in the single-index case, one-pass gradient-based algorithms can overcome CSQ performance dictated

by the Leap Coefficient [AAM22]. Our definition is consistent with the one of [Tro+25a].

- We prove that all polynomial multi-index functions are learned by Algorithm 5.1 either with total sample complexity $O(d)$ or $O(d\log(d)^2)$ if associated with the presence of a symmetry.
- The implementation of Algorithm 5.1 does not require pre-processing to initially correlate the estimation with the ground truth, and learns the meaningful representations from data alone. This is in contrast numerous findings (e.g., [MM18; LAL19; Mai+20] for single index targets or [CM20] for multi-index ones).
- We characterize the class of hard functions not learned in (almost) linear sample complexity. We show, however, that such functions can be learned through a hierarchical mechanism that extends the CSQ staircase first developed in [Abb+21] to a different, larger set, of functions.
- We validate the formal theoretical claims with detailed numerical illustrations. The code to reproduce representative figures is available in the project GitHub repository[1].

5.3 Single-Index Model

5.3.1 Main results

We first consider, for simplicity, the class of single-index models, in which $k = p = 1$. This corresponds to a mismatched setting of Generalized Linear Models, in which we want to learn

$$y = h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle), \quad \text{with} \quad f(\boldsymbol{z}, \boldsymbol{w}, a) = a\sigma(\langle \boldsymbol{w}, \boldsymbol{z}\rangle).$$

In this section we rigorously characterize the number of iterations needed for the class of Algorithms 5.1 to perform weak recovery (as in Definition 5.1) in the context of single-index targets. We establish a clear separation between the learning efficiency of the algorithmic family 5.1 and One-Pass SGD, limited by the Information Exponent of the target to be learned [BGJ21].

We start by introducing a restriction of the generative exponent in [Dam+24] to polynomial transformations.

Definition 5.2 (Polynomial Generative Exponent)**.** *We define $\ell_p^\star$ as the smallest integer k such that there exists a polynomial $p : \mathbb{R} \to \mathbb{R}$ with:*

$$\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[p(h^\star(x))H_k(x)\right] \neq 0, \tag{5.11}$$

Our assumptions on the SGD implementation and the target function are as follows:

Assumption 5.3. *Algorithm 5.1 is run with single-sample steps ($n_b = 1$), and uses the correlation loss*

$$\mathcal{L}(y, \hat{y}) = 1 - y\hat{y}.$$

[1] `https://github.com/IdePHICS/Repetita-Iuvant`

We also consider a spherical version of Algorithm 5.1, with

$$\boldsymbol{w}_{t+1} = \frac{\boldsymbol{w}_t - \gamma \nabla^{\perp}_{\boldsymbol{w}} \mathcal{L}\left(f(\boldsymbol{z}^\nu; \tilde{\boldsymbol{w}}_t(\rho), a_0), y^\nu\right)}{\left\| \boldsymbol{w}_t - \gamma \nabla^{\perp}_{\boldsymbol{w}} \mathcal{L}\left(f(\boldsymbol{z}^\nu; \tilde{\boldsymbol{w}}_t(\rho), a_0), y^\nu\right) \right\|} \tag{5.12}$$

where $\nabla^{\perp} f = \nabla f - \langle \nabla f, \boldsymbol{w} \rangle \boldsymbol{w}$ *is the spherical gradient.*

Assumption 5.4. *The activation* σ *is analytic almost everywhere, and* $\sigma^{(n)}$ *satisfies the polynomial growth assumption* (5.3) *for* $n \in \mathbb{N}$ *(possibly for different constants* C_n*). Further, for all* $n \geq 1$*, we have*

$$\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\sigma^{(n)}(x)\sigma'(x)^{n-1}\right] \neq 0 \quad \textit{and} \quad \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[x\sigma^{(n)}(x)\sigma'(x)^{n-1}\right] \neq 0$$

We are now ready to state our main result.

Theorem 5.2. *Suppose that Assumptions 5.1-5.4 hold, and let* $\rho = \rho_0 d^{-1}$*. Then, under an event of probability* $1/2$ *on the initialization* $(a_0, \boldsymbol{w}_0)$*, for each* $\delta > 0$*, there exist constants* $\gamma_0(\delta)$ *and* $C(\delta)$ *such that the following holds:*

- *If* $\ell_p^\star = 1$*, choosing* $\gamma = \gamma_0(\delta) d^{-1}$*, then* $\mathbb{P}(t_\eta^+ \leq C(\delta) \cdot d) \geq 1 - \delta$*,*
- *If* $\ell_p^\star = 2$*, choosing* $\gamma = \gamma_0(\delta)[d\log(d)]^{-1}$*, then* $\mathbb{P}(t_\eta^+ \leq C(\delta) \cdot d\log(d)^2) \geq 1 - \delta$*.*

The above holds for almost every choice of ρ_0 *under the Lebesgue measure. In particular, in a network with* $p = O(\log \frac{1}{\delta})$ *neurons, at least one of them achieves weak recovery with probability* $1 - \delta$*.*

Proof Sketch – We provide here the proof sketch for Theorem 5.2. For any vector $\boldsymbol{w}$, under the correlation loss, we have

$$\nabla_{\boldsymbol{w}} \mathcal{L}(f(\boldsymbol{z}; \boldsymbol{w}, a_0), y) = -a_0 y \sigma'(\langle \boldsymbol{w}, z \rangle) \cdot \boldsymbol{z}, \tag{5.13}$$

so the gradient is aligned with $\boldsymbol{z}$. As a result, we have

$$\nabla_{\boldsymbol{w}} \mathcal{L}\left(f(\boldsymbol{z}; \tilde{\boldsymbol{w}}(\rho), a_0), y\right) = -a_0 y \sigma'\left(\langle \boldsymbol{w}, \boldsymbol{z} \rangle + a_0 \rho y \sigma'(\langle \boldsymbol{w}, \boldsymbol{z} \rangle) \cdot \underbrace{\|\boldsymbol{z}\|^2}_{\approx d}\right) \cdot \boldsymbol{z} \tag{5.14}$$

This expression for the gradient exhibits two important properties:

- Even though $\rho \asymp d^{-1}$, since the gradient lies along $\boldsymbol{z}$, the additional dot product with $\boldsymbol{z}$ amplifies the signal by a factor of d,
- The resulting gradient (5.14) is a *non-linear* function of y, as opposed to the linear one of (5.13).

The latter property enables us to show that the dynamics driven by equation (5.14) can implement all polynomial transformations of the output y, for almost all choices of ρ_0. Subsequently, the SGD algorithm on the transformed input $p(y)$ can be studied with the same techniques as [BGJ21], but with an information exponent equal to $\ell_p^\star$.

5.3.2 Discussion

Significance and necessity of the assumptions – Assumption 5.3 simplifies Algorithm 5.1 to a more tractable version. In particular, the normalization step allows us to keep track of only one parameter, the overlap between $\boldsymbol{w}_t$ and $\boldsymbol{w}^\star$. We expect however Theorem 5.2 to hold in the general setting of Algorithm 5.1.

Assumption 5.4 is a regularity Assumption, ensuring that some quantities of interest in the proof are non-zero. The first part is satisfied by virtually every activation function; the second is more restrictive, but we show that it is satisfied for biased activations:

Lemma 5.1. *Let σ be a non-polynomial analytic function. Then the second part of Assumption 5.4 is true for the function $x \mapsto \sigma(x+b)$ for almost every choice of b (according to the Lebesgue measure).*

Finally, although we state our result as an almost sure event over ρ_0, it can be reformulated into an event on the second layer weight a_0, in which the randomness comes from the initialization step.

Learning polynomial functions – Although our definition of the Polynomial Generative Exponent is more restrictive than the one in [Dam+24], we show that it is enough to allow learning of a large class of functions, namely polynomials:

Theorem 5.3. *Assume that $h^\star$ is a polynomial function. Then the Polynomial Generative Exponent of $h^\star$ is always at most 2, and is equal to 1 whenever $h^\star$ is not even.*

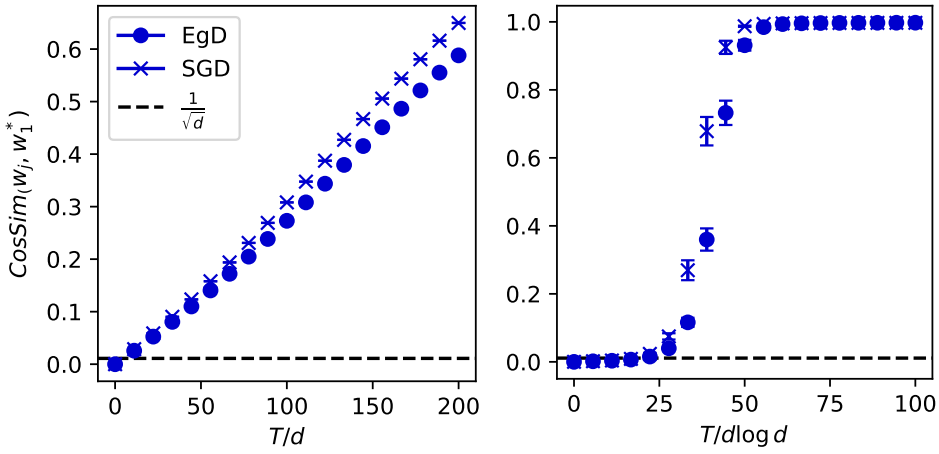
This theorem implies that the Extragradient algorithm class allows us to match the sample complexity of the algorithm of [CM20], without the need for *ad hoc* preprocessing.

Beyond Algorithm 5.1 – A succinct summary of the proof goes as follows: the first time we see a sample $\boldsymbol{z}$, we store information about y along the direction of $\boldsymbol{z}$. The second time we see the same sample, the component along $\boldsymbol{z}$ in $\boldsymbol{w}$ interact *non-linearly* with y, which bypasses the CSQ framework. Importantly, we expect that this behavior also appears in SGD algorithms where we do not see the data twice *in a row*, but *across multiple epochs*! Indeed, upon small enough correlation between the $\boldsymbol{z}^\nu$, seeing multiple examples before the repetition of $\boldsymbol{z}$ should not interfere with the data stored along this direction. We are therefore not claiming that Algorithm 5.1 is superior to other classically-used methods; rather, it provides a tractable and realistic setting to study data repetition in SGD, which was ignored in previous theoretical works on the topic.

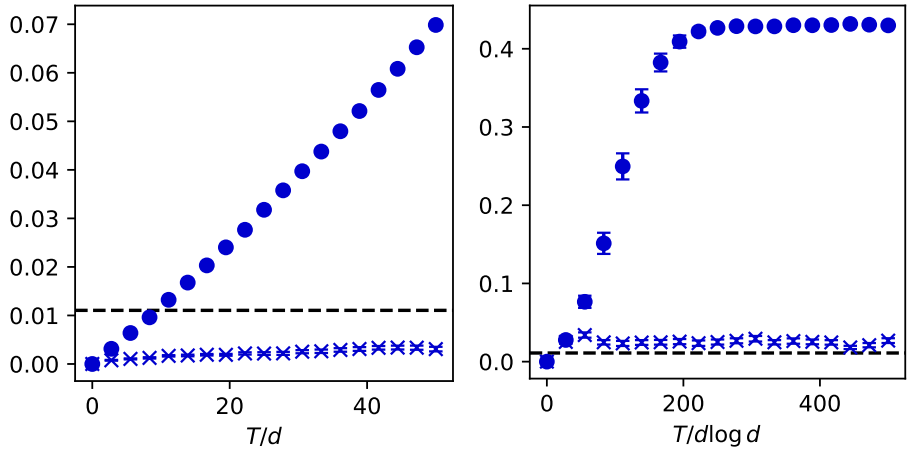
5.3.3 Numerical illustrations

We illustrate in Figure 5.1 the stark difference between the weak recovery dynamics of One-Pass SGD and Algo. 5.1 for single-index targets. The Cosine Similarity between the learned weight $\boldsymbol{w}_t$ and the ground truth informative direction $\boldsymbol{w}_\star$ is shown as a function of the time steps t for different target functions. Two optimization routines are considered to exemplify the learning behavior:

(i) Extragradient Descent (EgD), corresponding to the family of Algorithms 5.1 with positive ρ parameters;



(a) $h^\star(x) = \mathrm{relu}(x)$ (b) $h^\star(x) = \mathrm{He}_2(x)$



(c) $h^\star(x) = \mathrm{He}_3(x)$ (d) $h^\star(x) = \mathrm{He}_4(x)$

Figure 5.1: **Learning single-index targets** – Evolution of the Cosine Similarity attained by EgD (dots) and SGD (crosses) as a function of the normalized iteration time. (dashed horizontal line $\frac{1}{\sqrt{d}}$ is a visual guide to place random performance). **(a)** $(\ell, \ell^\star) = (1, 1)$**:** both algorithms learn in linear time. **(b)** $(\ell, \ell^\star) = (2, 2)$**:** both algorithms learn in $O(d \log d)$ time. **(c)** $(\ell, \ell^\star) = (3, 1)$**:** EgD learns in linear time, while SGD requires $O(d^2)$ time. **(d)** $(\ell, \ell^\star) = (4, 2)$**:** EgD learns in $O(d \log d)$ time, SGD requires $O(d^3)$ time (Details in Appendix E.5).

(ii) One-Pass SGD, corresponding to vanilla SGD, or equivalently Algorithm 5.1 associated with $\rho = 0$ hyperparameter.

The scaling of (γ, ρ) as a function of the input dimension d and exponent $\ell^\star$ are given in Theorem 5.2, while the mini-batch size is fixed to $n_b = 1$ (See Appendix E.4 for extension to larger n_b). For all plots, we take $\sigma = \text{relu}$ and we refer to Appendix E.5 for details.

- **SGD-easy non-symmetric targets ($\ell = \ell^\star = 1$)** – The Left section of Figure 5.1 shows $h^\star(x) = \text{relu}(x)$. Here both SGD and Algorithm 5.1 learn in $T = O(d)$.
- **SGD-easy symmetric targets ($\ell = \ell^\star = 2$)** – The Center-Left section of Figure 5.1 shows $h^\star(x) = \text{He}_2(x)$. Here both SGD and Algorithm 5.1 learn in $T = O(d \log d)$.
- **SGD-hard non-symmetric targets ($\ell > 2, \ell^\star = 1$)** – The Center-Right section of Figure 5.1 shows $h^\star(x) = \text{He}_3(x)$. Here Algorithm 5.1 learns in $T = O(d)$ while One Pass SGD suffers from the limitations detailed in Equation (5.9), i.e. $T = O(d^{\ell-1})$ with $\ell = 3$ for this case.
- **SGD-hard symmetric targets ($\ell > 2, \ell^\star = 2$)** – The Right section of Figure 5.1 shows $h^\star(x) = \text{He}_4(x)$. Here Algorithm 5.1 learns in $T = O(d \log d)$ while One Pass SGD suffers from the time scaling law detailed in Equation (5.9) $T = O(d^{\ell-1})$.

5.4 Multi-Index Model

We now investigate the superiority of Algorithm 5.1 over the idealized One-Pass SGD training scheme when learning multi-index targets. In this setting, the networks weights can align with only a few select directions from $V^\star$. To quantify this behavior, we define directional versions of the Information and Generative Exponents (see also [Tro+25a]):

Definition 5.3. *Let $\boldsymbol{v} \in V^\star$. The Information Exponent $\ell_{\boldsymbol{v}}$ (respectively* Polynomial Generative Exponent *$\ell^\star_{\boldsymbol{v}}$) of y in the direction $\boldsymbol{v}$ is the smallest k such that*

$$\mathbb{E}\left[yH_k(\langle \boldsymbol{v}, \boldsymbol{z}\rangle)\right] \neq 0 \quad (\textit{resp.}\mathbb{E}\left[p_{\boldsymbol{v}}(y)H_k(\langle \boldsymbol{v}, \boldsymbol{z}\rangle)\right] \neq 0)$$

for a polynomial $p_{\boldsymbol{v}}$. We also let $\ell = \min_{\boldsymbol{v}} \ell_{\boldsymbol{v}}$ and $\ell^\star_p = \min_{\boldsymbol{v}} \ell^\star_{\boldsymbol{v}}$. The minimal subspace $U^\star \subseteq V^\star$ is defined as the smallest subspace U such that $\ell_{\boldsymbol{v}} > \ell$ (resp. $\ell^\star_{\boldsymbol{v}} > \ell^\star_p$) for every $v \in U^\perp$.

The identification of the class of hard functions has been subject of intense theoretical scrutiny [Abb+21; AAM22; AAM23; Dan+24a; BBP23; Tro+25a]; and for the problem of initial alignment it was shown to depend on the Information exponent defined above. Thus, similarly to the single-index scenario, the time complexity needed for Algorithm 5.1 to weakly recover the target subspace follows the law in Equation (5.9) for the multi-index information exponent ℓ. A natural question is thus to ask whether the class of Algorithm 5.1 also allows us to bypass this requirement. We show that this is indeed the case. Consider a neural network as in Equation (5.1); we assume that the internal learning rate ρ is chosen independently for each neuron:

Assumption 5.5. *For each $j \in [p]$, the parameter $\rho_{0,j}$ is chosen independently from all other neurons, according to a* symmetric *distribution which is absolutely continuous w.r.t the Lebesgue measure.*

Theorem 5.4. *Let $\delta > 0$, and $\Pi^\star$ the projection operator on $V^\star$. There exists constants $\gamma_0(\delta), C(\delta)$ depending only on δ such that if we take $p \geq C(\delta)k$ and*

- *if $\ell_p^\star = 1$, $\gamma = \gamma_0(\delta)d^{-1}$ and $T = C(\delta)d$,*
- *if $\ell_p^\star = 2$, $\gamma = \gamma_0(\delta)(d\log(d))^{-1}$ and $T = C(\delta)d\log(d)^2$,*

then with probability $1-\delta$, there exists a $t \leq T$ such that at least a positive proportion of the vectors $\boldsymbol{w}_{t,i}$ satisfy $\left\|\Pi^\star \boldsymbol{w}_{t,i}\right\| \geq \eta$. Further, if $\ell_p^\star = 1$, then the vectors $\{\Pi^\star \boldsymbol{w}_{t,i}\}_{i\in[p]}$ span the minimal subspace $U^\star$.

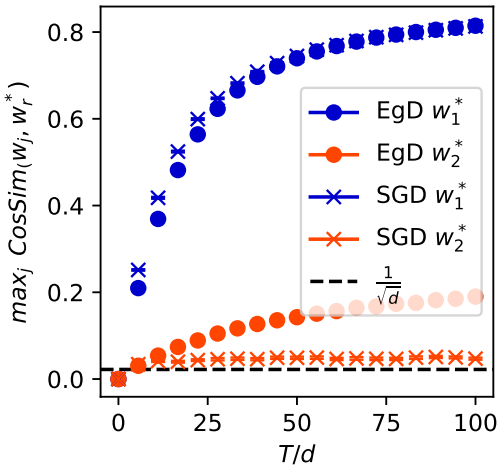
5.4.1 Numerical investigation with multi-index target

We illustrate the results of Theorem 5.4 numerically by comparing the performance of One-Pass SGD with Algorithm 5.1 in Figure 5.3. We measure the maximal cosine similarity between the hidden layer neurons and the different rows of the target's weight matrix $W^\star$ as a function of the iteration time. Furthermore, we set without loss of generality the target's directions to the standard basis of $\mathbb{R}^d$ to ease the notation. The algorithms considered are again EgD and vanilla One-Pass SGD (See Appendix E.5 for details).

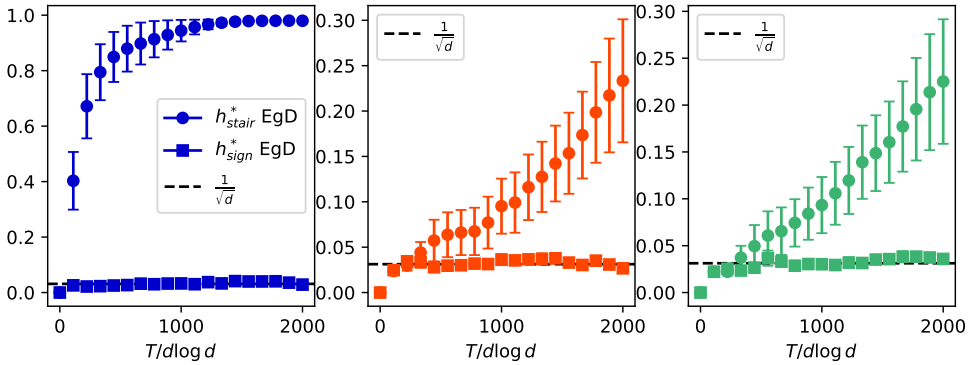
- **SGD-easy non-symmetric targets ($\ell = \ell^\star = 1$)** – Figure 5.3, left, shows $h^\star(x_1, x_2) = x_1 + x_1x_2$. Here both SGD and Algorithm 5.1 learn in $T = O(d)$ the directions $(\boldsymbol{w}_1^\star, \boldsymbol{w}_2^\star)$.
- **SGD-easy symmetric targets ($\ell = \ell^\star = 2$)** – Figure 5.3, center left, shows $h^\star(x_1, x_2) = x_1x_2$. Here both SGD and Algorithm 5.1 learn in $T = O(d\log d)$ the directions $(\boldsymbol{w}_1^\star, \boldsymbol{w}_2^\star)$.
- **Non-symmetric targets with SGD-easy and hard directions ($\ell_{\boldsymbol{v}} > 2, \ell_{\boldsymbol{v}}^\star = 1$)** – Figure 5.3, center-right, shows $h^\star(x_1, x_2) = x_1 + \mathrm{He}_3(x_2)$. Here both SGD and Algorithm 5.1 learn in $T = O(d)$ the direction $\boldsymbol{w}_1^\star$, that satisfies $\ell_{\boldsymbol{w}_1^\star} = \ell_{\boldsymbol{w}_1^\star}^\star = 1$. However, since $\ell_{\boldsymbol{w}_2^\star} = 3$ while $\ell_{\boldsymbol{w}_2^\star}^\star = 1$, One-Pass SGD suffers from the limitations detailed in Equation (5.9) and requires $\Omega(d^2)$ samples to learn the second direction $\boldsymbol{w}_2^\star$. This contrasts with the behavior of Algorithm 5.1 that learns also $\boldsymbol{w}_2^\star$ in $T = O(d)$ steps.
- **SGD-hard symmetric targets ($\ell > 2, \ell^\star = 2$)** – The right section Figure 5.3 shows $h^\star(x_1, x_2, x_3) = x_1x_2x_3$. The Information Exponent is $\ell = 3$, hence vanilla SGD does not learn any direction in the target subspace in $T = O(d\log d)$ iterations. However, Algorithms 5.1 learns in $T = O(d\log d)$ steps all the three directions $\{\boldsymbol{w}_1^\star, \boldsymbol{w}_2^\star, \boldsymbol{w}_3^\star\}$ since the Generative Exponents $\ell_{\boldsymbol{w}_r^\star}^\star$ are all equal to two.

5.4.2 Learning hard functions through a hierarchical mechanism

The investigation of the hierarchical nature of SGD learning has attracted noticeable attention [Abb+21; AAM22; AAM23; Dan+24a]. In this paper, we portray a completely different picture in terms of computational efficiencies when data repetition is considered in the algorithmic SGD routine. One may wonder if there is a generalization of such a hierarchical learning mechanism to the present novel setting. Indeed for optimal AMP algorithm, a different picture emerges, (called *Grand staircase* in [Tro+25a]). Here we show that coupling between directions can hierarchically guide the learning process as well with SGD.

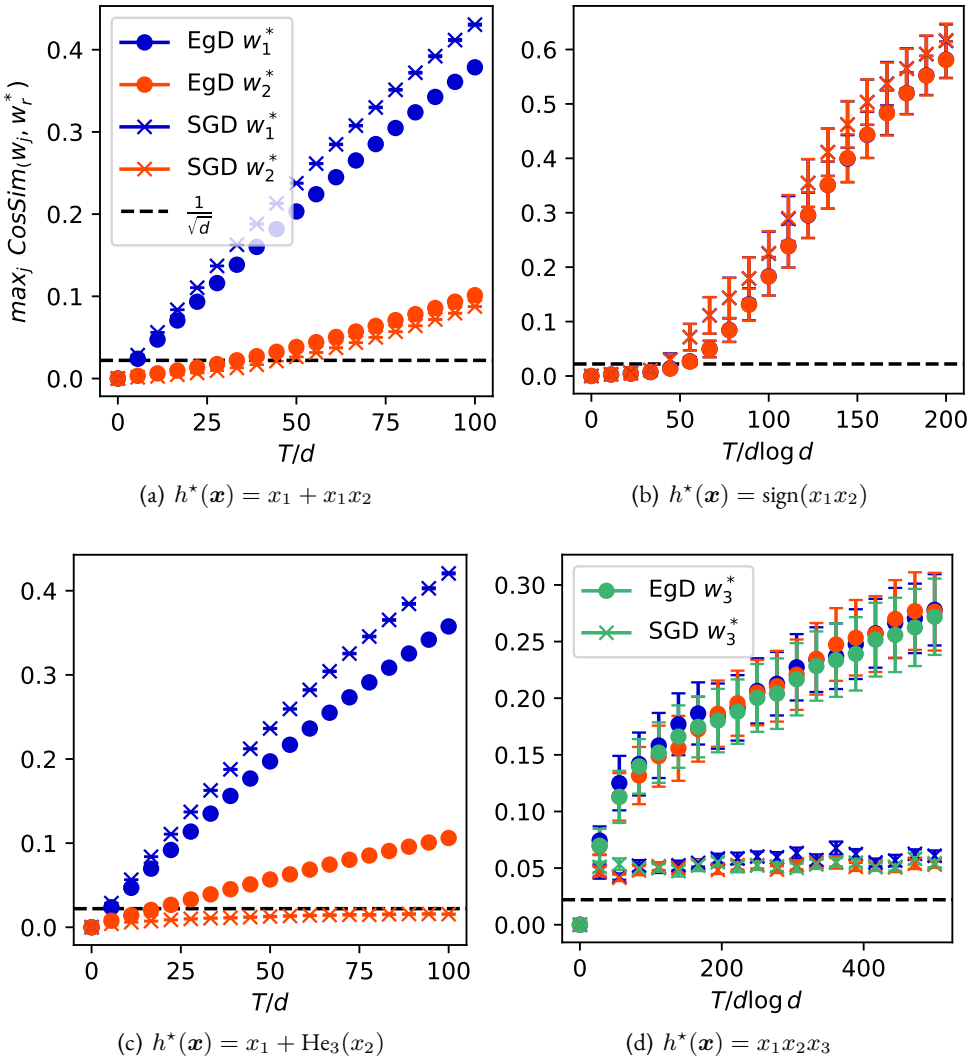


(a) $h^\star(\boldsymbol{x}) = x_1 + x_1 \mathrm{He}_3(x_2)$



(b) $h^\star_{\text{sign}}(\boldsymbol{x}) = \text{sign}(x_1x_2x_3), \quad h^\star_{\text{stair}}(\boldsymbol{x}) = \mathrm{He}_2(x_1) + \text{sign}(x_1x_2x_3)$

Figure 5.2: **Hierarchical learning** – Evolution of the maximum Cosine Similarities with different target directions $\{\boldsymbol{w}_r^\star\}_{r\in[k]}$ are identified by different colors: $\boldsymbol{w}_1^\star$ (blue), $\boldsymbol{w}_2^\star$ (orange), $\boldsymbol{w}_3^\star$ (green) as a function of the normalized iteration time. **(a):** both the algorithms learn $\boldsymbol{w}_1^\star$ in linear time, but only EgD can also learn $\boldsymbol{w}_2^\star$ in $O(d)$ via a staircase mechanism; SGD requires another $O(d^2)$ steps to take advantage of the staircase and learn $\boldsymbol{w}_2^\star$. **(b):** EgD performance for the target function $f^\star_{\text{stair}}$ (dots) and $f^\star_{\text{sign}}$(squares); we plot in separate subplots the overlap with different directions to highlight the presence of hierarchical learning mechanism. Learning the first target direction $\boldsymbol{w}_1^\star$ triggers the hierarchical mechanism and EgD is able to weakly recover the full target subspace $\text{Span}(\boldsymbol{w}_1^\star, \boldsymbol{w}_2^\star, \boldsymbol{w}_3^\star)$ in $O(d \log d)$, while this does not happen when removing $\mathrm{He}_2(x_1)$ from the target. See Appendix E.5 for additional details.



(a) $h^\star(\boldsymbol{x}) = x_1 + x_1x_2$ (b) $h^\star(\boldsymbol{x}) = \text{sign}(x_1x_2)$

(c) $h^\star(\boldsymbol{x}) = x_1 + \text{He}_3(x_2)$ (d) $h^\star(\boldsymbol{x}) = x_1x_2x_3$

Figure 5.3: **Learning multi-index targets** – Evolution of the maximum Cosine Similarities attained by EgD (crosses) and SGD (dots) as a function of the normalized iteration time. The Different directions $\{\boldsymbol{w}_r^\star\}_{r\in[k]}$ are identified by colors: $\boldsymbol{w}_1^\star$ (blue), $\boldsymbol{w}_2^\star$ (orange), $\boldsymbol{w}_3^\star$ (green). (dashed horizontal line $\frac{1}{\sqrt{d}}$ is a visual guide to place random performance). **(a)** $(\ell, \ell^\star) = (1, 1)$**:** both algorithms learn the first direction in linear time, as well as the second one in $O(d)$ steps using the staircase mechanism. **(b)** $(\ell, \ell^\star) = (2, 2)$**:** both algorithms learn the two directions simultaneously in $O(d \log d)$ steps. **(c)** $(\ell_{\boldsymbol{w}_2^\star}, \ell^\star_{\boldsymbol{w}_2^\star}) = (3, 1)$**:** both algorithms learn $\boldsymbol{w}_1^\star$ in linear time, but only EgD learns $\boldsymbol{w}_2^\star$ in $O(d)$; SGD instead requires $O(d^2)$. **(d)** $(\ell, \ell^\star) = (3, 2)$**:** EgD learns all 3 directions simultaneously in $O(d \log d)$ steps, while SGD needs $O(d^2)$ time (details in Appendix E.5).

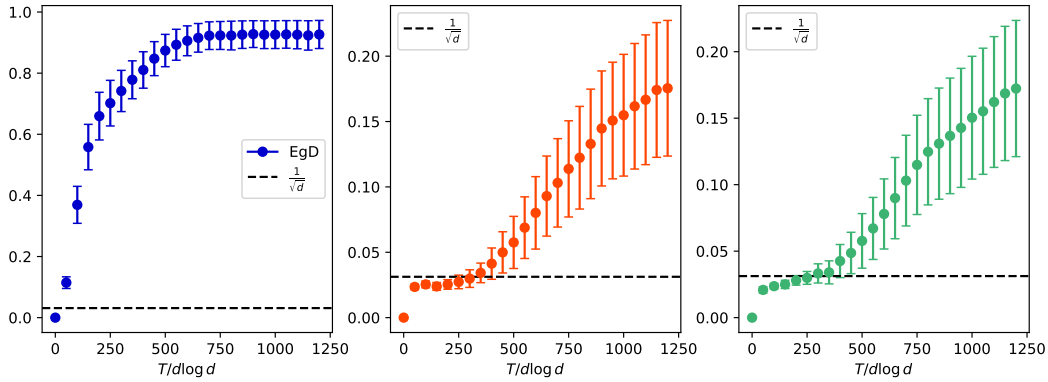


Figure 5.4: **Hierarchical learning** – Evolution of the maximum Cosine Similarities attained by EgD with different target directions $\{\boldsymbol{w}_r^\star\}_{r\in[k]}$ are identified by different colors: $\boldsymbol{w}_1^\star$ (blue), $\boldsymbol{w}_2^\star$ (orange), $\boldsymbol{w}_3^\star$ (green) as a function of the normalized iteration time. The target function is $f^\star_{\text{sq}-\text{stair}}(\boldsymbol{z}) = \text{He}_4(z_1) + \text{sign}(z_1z_2z_3)$, that is an SQ-staircase (or grand staircase), not a CSQ-staircase. EgD learn $\boldsymbol{w}_1^\star$ in $O(d\log d)$ steps and use the information to learn the other two directions in another $O(d\log d)$ steps. SGD (not showed in the plot) cannot take advantage of the staircase mechanism since $\text{He}_4(z_1)$ and $\text{sign}(z_1z_2z_3)$ have information exponent $\ell = 4$ and $\ell = 3$ respectively. (See Appendix E.5).

The Left panel in Figure 5.2 shows an example of the so-called staircase functions [Abb+21], i.e. $f^\star(\boldsymbol{z}) = z_1 + z_1\text{He}_3(z_2)$. While one-pass SGD needs to learn hierarchically first the direction $\boldsymbol{w}_1^\star$ in $T = O(d)$ and then $\boldsymbol{w}_2^\star$ in $T = O(d^2)$, Algorithm 5.1 easily learns both in linear time. This observation could lead to infer that hierarchical learning mechanisms are not present when more realistic training scenarios are considered.

To refute this, we illustrate in the right panel of Figure 5.2 a scenario that precisely exemplifies the presence of hierarchical mechanisms within Algorithm 5.1. We run this algorithm on two different target functions, $f^\star_{\text{sign}}(\boldsymbol{z}) = \text{sign}(z_1z_2z_3)$ and $f^\star_{\text{stair}}(\boldsymbol{z}) = \text{He}_2(z_1) + \text{sign}(z_1z_2z_3)$. We show in Appendix E.2 that 3-sparse parities of the form $\text{sign}(z_1z_2z_3)$ are hard functions even for SGD with data repetition (Algorithm 5.1), and indeed they are not learned in (almost) linear time. On the other hand, for the function $f^\star_{\text{stair}}$, our simulations predict that the first direction $\boldsymbol{w}_1^\star$ is learned in $T = O(d\log d)$ iterations (see the rightmost panel of Figure 5.3). However, once the direction $\boldsymbol{w}_1^\star$ is learned, it can be used to obtain order-one correlation with $\{\boldsymbol{w}_2^\star, \boldsymbol{w}_3^\star\}$ again in $T = O(d\log d)$ steps. While the latter example belongs to the class of "CSQ" staircase function depicted in [Abb+21], novel "SQ" hierarchical mechanisms arise in our framework, such as for instance the function $f^\star_{\text{sq}-\text{stair}}(\boldsymbol{z}) = \text{He}_4(z_1) + \text{sign}(z_1z_2z_3)$ (see Figure 5.4).

The staircase picture in terms of generative exponent was discussed for AMP-type algorithm in [Tro+25a]. Our study indicate that this new class (they called grand staircase) is not restricted to AMP: We anticipate that grand staircase functions can be efficiently learned by two-layer networks when data reusing is permitted, whereas only standard staircase ones emerge under single-pass SGD. Additional recent results by [JMS24] align with this perspective.

5.5 Conclusions

We have pushed in this paper the boundaries of learning multi-index models in high-dimensional settings using gradient-type algorithms, and shown that, contrary to what was believed, they are very efficient in doing so. We argue in this manuscript with a simple analytically solvable model how to bypass the limitations of the CSQ framework: we process the same sample twice. Crucially, a similar phenomenon appears also in standard SGD when training over multiple epochs, with the only difference being that the same data is not seen consecutively. We believe that the model proposed in this manuscript paves the way for the analysis of more realistic training scenario where *correlation* in the data is taken into account, surpassing the limiting i.i.d. scenario.

Upon completion of this work, we became aware of the interesting paper from [Lee+24] showing strong recovery with similar sample complexity, also by exploiting the reuse of training data. Our work was conducted independently and simultaneously

Acknowledgements

The authors would like to thank Nicolas Flammarion, Bruno Loureiro, Nati Srebro, Emanuele Troiani, and Lenka Zdeborová for interesting discussions. This work was supported by the Swiss National Science Foundation under grant SNSF OperaGOST (grant number 200390).

Chapter 6

Asymptotics of SGD in Sequence-Single Index Models and Single-Layer Attention Networks

Luca Arnaboldi, Bruno Loureiro, Ludovic Stephan, Florent Krzakala, and Lenka Zdeborová

Abstract. We study the dynamics of stochastic gradient descent (SGD) for a class of sequence models termed Sequence Single-Index (SSI) models, where the target depends on a single direction in input space applied to a sequence of tokens. This setting generalizes classical single-index models to the sequential domain, encompassing simplified one-layer attention architectures. We derive a closed-form expression for the population loss in terms of a pair of sufficient statistics capturing semantic and positional alignment, and characterize the induced high-dimensional SGD dynamics for these coordinates. Our analysis reveals two distinct training phases: escape from uninformative initialization and alignment with the target subspace, and demonstrates how the sequence length and positional encoding influence convergence speed and learning trajectories. These results provide a rigorous and interpretable foundation for understanding how sequential structure in data can be beneficial for learning with attention-based models.

Stochastic Gradient Descent (SGD) is the core optimization tool driving modern machine learning. Recent years have seen substantial progress in understanding its dynamics, particularly in two-layer networks [SS95b; MMN18; CB18; RV22; SS20; Arn+23]. While global convergence is qualitatively well-understood when the network is wide enough, quantitative results are scarcer. A particularly fruitful body of recent theoretical work addressing this gap has focused on deriving precise convergence rates for particular model classes on synthetic data, such as high-dimensional Gaussian single and multi-index models [BGJ21; AAM22; AAM23]. These advances have sparked a wave of follow-up studies [DLS22; Dam+23; BBP23; Ba+23; Mon+24; Mou+23; Cui+24b; ZB24; BMZ24; Arn+24a; Arn+24b], deepening our understanding of what problems are hard to learn for neural networks trained under SGD.

While multi-index models have served as a cornerstone in theoretical analyses of learning, they remain far from the architectures driving recent breakthroughs in machine learning. Modern advances in learning from sequential data — particularly in natural language processing — are increasingly dominated by attention-based models such as the Transformer architecture [Vas+17]. These models introduce a paradigm shift through self-attention mechanisms, which dynamically reweight the influence of each input token based on its relevance to others. Through successive layers of attention, Transformers capture intricate dependencies across sequences, enabling state-of-the-art performance in tasks ranging from machine translation to large-scale language modeling [Bro+20; Dev+18].

The main goal of this work is to extend our theoretical understanding of learning with SGD on multi-index models to attention-based architectures and sequential data. Inspired by [Cui+24a; Tro+25b], our focus will be on the following class of single-layer, tied attention models:

$$\boldsymbol{f}_{\boldsymbol{w}}(X) = \Pi\left[\text{softmax}\left(\left(X + \frac{P}{\sqrt{d}}\right)\boldsymbol{w}\boldsymbol{w}^\top\left(X + \frac{P}{\sqrt{d}}\right)^\top\right)\right], \tag{6.1}$$

where $\boldsymbol{w} \in \mathbb{R}^d$ are the trainable weights and Π is the *reduction map*, which allows passing from an $L \times L$ matrix to a k_{out}-dimensional vector. As detailed in Appendix F.1, this model is a reduction of the standard attention mechanism, where

(i) key and query matrix are tied, and $d_{\text{head}} = 1$: $Q = K = X \cdot \boldsymbol{w} \in \mathbb{R}^{L\times 1}$;

(ii) since we are considering a single layer attention, and we do not need to learn a new representation of the sequence, the value matrix is the identity: $V = I_L$.

In this work, we will be interested in the optimization properties of the model in Equation (6.1) when trained under (spherical) one-pass stochastic gradient descent (SGD) from a random initial condition $\boldsymbol{w}^0 \sim \text{Unif}(\mathbb{S}^{d-1}(\sqrt{d}))$:

$$\boldsymbol{w}^{\nu+1} = \frac{\boldsymbol{w}^\nu - \gamma\nabla_{\boldsymbol{w}}\ell(X^\nu, \boldsymbol{y}^\nu; f_{\boldsymbol{w}})}{\left\|\boldsymbol{w}^\nu - \gamma\nabla_{\boldsymbol{w}}\ell(X^\nu, \boldsymbol{y}^\nu; f_{\boldsymbol{w}})\right\|}\|\boldsymbol{w}^\nu\|\,. \tag{6.2}$$

with the squared loss:

$$\ell(X, \boldsymbol{y}; f_{\boldsymbol{w}}) = \left\|\boldsymbol{y} - f_{\boldsymbol{w}}(X)\right\|_{\text{F}}^2 = \sum_{i=1}^{k_{\text{out}}}\left(y_i - f_{\boldsymbol{w}}(X)_i\right)^2. \tag{6.3}$$

Note that for one-pass (a.k.a. *online* or *streaming*) SGD each sample is only seen once, meaning that the sample complexity of the algorithm coincides with the convergence rate. The spherical constraint is considered to simplify the mathematical analysis, a common assumption in the analysis of SGD for single-index models [BGJ21; DLS22].

To derive a sharp characterization of the sample complexity and convergence rate of SGD for the single-layer attention mechanism in Equation (6.1), we assume that the sequence data $(X, \boldsymbol{y})$ is generated from the following Gaussian *sequence single-index* (SSI) model:

Assumption 6.1 (Data distribution)**.** *We assume training data* $(X, \boldsymbol{y}) \in \mathbb{R}^{L\times d} \times \mathbb{R}^{k_{\text{out}}}$ *is independently drawn from a Gaussian* Sequence Single Index (SSI) *model:*

$$\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}(X) = \boldsymbol{g}(X \cdot \boldsymbol{w}_\star) \tag{6.4}$$

where $X \in \mathbb{R}^{L\times d}$ *is a Gaussian matrix with entries* $\mathcal{N}(0, 1/d)$*,* $\boldsymbol{g}\colon \mathbb{R}^L \to \mathbb{R}^{k_{\text{out}}}$ *is a vector-valued link function that depends only on the scalar product of the input with a fixed vector* $\boldsymbol{w}_\star \in \mathbb{S}^{d-1}(\sqrt{d})$ *and* k_{out} *is an integer that does not depend on* d *or* L*.*

Recent work by [Cui+24a; Tro+25b; Cui25] has shown that the single-layer tied attention model in Equation (6.1) is a particular instance of a class of *sequence multi-index (SMI) models*, creating a bridge between single-index analysis and attention-based learning. This mapping implies that model Equation (6.1) can learn at best a predictor in this class, justifying the choice for training data. The SSI model class was introduced by [Cui+24a] in the context of studying phase transitions for the attention mechanism. These results position the SMI model as a signature synthetic model to study the interplay between attention mechanisms, data structure, and the dynamics of learning algorithms. Despite this progress, the learning dynamics of SMI models under SGD remain unexplored. Prior studies [Cui25; Cui+24a] analyzed the empirical risk minimizer through the heuristic replica method, while [Tro+25b] rigorously studied the Bayes-optimal estimator for SMI models. However, a theoretical understanding of the population landscape and SGD dynamics in this model is missing. Our work addresses this gap, providing the first rigorous characterization of SGD dynamics in SMI models by leveraging techniques developed for single and multi-index models.

Main results — Our main methodological contribution is the generalization of analytical tools to study multi-index models to variants that process sequences of tokens rather than simple vector inputs. Our contributions can be summarized as follows:

- We introduce the notion of *sequence information exponent* (SIE), as a generalization of the information exponent for single-index models [BGJ21]; the SIE has a direct correspondence with the sample complexity of SGD. We also discuss the implications of the positional encoding on the sample complexity, proving it could help SGD to learn faster.
- We analyze the speed-up introduced by the attention mechanism when learning sequential data, compared to models not adapted to treat sequence structures, e.g. fully connected networks. For many problems, we show that the gain is proportional to the sequence length L and in some cases even larger.
- We investigate the interplay between the positional and semantic structure of the data following the setting from [Cui+24a], showing that SGD dynamics is not always able to disentangle the two and that a rich phase diagram arises describing the structure of the corresponding population loss and the performance of SGD.

All our formal claims are supported by rigorous proofs, as well as numerical experiments; the code developed is available at this GitHub repository[1].

[1] `https://github.com/IdePHICS/Sequence-Single-Index`

Further Related works — There has been much activity discussing **SGD with synthetic data on multi-index models** over the last decades or so, see e.g. [BGJ21; Vei+22; Arn+23; Col+24; MB23; BMZ24; MU25] and references therein. **Information** and **generative** exponent have been the topic of many works over the last few years [BGJ21; AAM22; Dam+24; Tro+25a; Def+25]. Here we discuss and adapt these notions for sequence models [Tro+25b].

On the theory of transformers, [SHT23] has shown that transformers can approximate certain sparse averaging (qSA) tasks independently of the sequence length, while fully connected networks require a width scaling with the length. [Mou+25] derived sample complexity upper bounds for q-Sparse Token Regression (qSTR) tasks, a random version of qSA. [Wan+24] showed that the separation between transformers and fully-connected networks also hold under GD for the qSTR task. While the qSA and qSTR tasks are not directly comparable to the SSI model, both model a sparse dependency of the task in the sequence. [Mar+24] introduced a model that can be seen as a sequence two-index model. While we assume the input data to be i.i.d. Gaussian they assume a spiked covariance structure in the data which makes their results not directly comparable to ours.

Convergence guarantees for a single layer transformer were studied by [Wu+23]. [Son+24] studied the convergence of GD for simple architectures and highlighted the existence of suboptimal local solutions. [Li+25] point out that rapid convergence does not guarantee meaningful learning. [Li+23] give sample complexity bounds for a shallow vision transformer. [Zha+25] studied benign overfitting in transformers trained by SGD. [YF25] derived sample complexity upper bounds for next token prediction tasks based on the complexity of the hypothesis class, and showed that the out-of-sample error depends on the sequence mixing properties. Compared to these work we move beyond convergence results to study how the data structure, e.g. sequence structure and positional encodings, affect sample complexity and behaviors of the population loss and recovery dynamics in the high-dimensional limit.

6.1 Setting and definitions

Let $(X^\nu, \boldsymbol{y}^\nu) \in \mathbb{R}^{L\times d} \times \mathbb{R}^{k_{\text{out}}}$ denote $\nu = 1, \dots, n$ samples drawn from the Gaussian sequence single-index model with weights $\boldsymbol{w}_\star \in \mathbb{R}^d$ and link function $\boldsymbol{g}$, defined in Equation (6.4). As motivated in the introduction, our goal in this work is to characterize the sample complexity of learning the sequence task $(X^\nu, \boldsymbol{y}^\nu)$ with a tied single-layer attention trained under one-pass (spherical) SGD defined in Equation (6.2). Note that while the model might appear as too simplified because of the lack of correlation between the tokens, we will show that it is sufficient to capture the main features of sequence models.

As shown in the classical result by [Chu54], one-pass SGD can be understood as a noisy discretization of gradient flow on the *population risk* (also referred to as *population loss*):

$$R(\boldsymbol{w}) = \mathbb{E}_{X\sim\mathcal{N}(0,I_d/d)}\left[\ell(X, \boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}(X); \boldsymbol{f}_{\boldsymbol{w}})\right]. \tag{6.5}$$

Therefore, in order to understand the dynamics in Equation (6.2) it is important to understand the landscape of the risk above. The key property of single-index models underlying the convergence rate analysis of [BGJ21] is that rotation invariance of the population risk implies that it

only depends on a single parameter: the correlation between the target weights and the predictor weight, also known as an *order parameter* or *sufficient statistic*. A similar property holds for the family of SSI models defined by Assumption 6.1. Indeed, conditionally on the weights, the following projections

$$\boldsymbol{z}_\star = X \cdot \boldsymbol{w}_\star \in \mathbb{R}^L \quad \text{and} \quad \boldsymbol{z} = \left(X + \frac{P}{\sqrt{d}}\right) \boldsymbol{w} \in \mathbb{R}^L, \tag{6.6}$$

define joint Gaussian variables, which can be fully characterized by their means and covariances:

$$\begin{gathered} \mathbb{E}\left[z_{\star,i}\right] = 0, \quad \mathbb{E}\left[z_i\right] = e_i \\ \operatorname{Cov}\left(z_{\star,i}, z_{\star,j}\right) = \delta_{ij}, \quad \operatorname{Cov}\left(z_i, z_j\right) = \delta_{ij}, \quad \operatorname{Cov}\left(z_{\star,i}, z_j\right) = \delta_{ij} m, \end{gathered} \tag{6.7}$$

where we have introduced the *sufficient statistics*:

$$m = \frac{\boldsymbol{w}_\star^\top \boldsymbol{w}}{d} \quad \text{and} \quad \boldsymbol{e} = \frac{P\boldsymbol{w}}{\sqrt{d}}. \tag{6.8}$$

These play exactly the same role as the overlap between target and predictor weights in the standard single-index model. Therefore, the population risk can be written as a function of these statistics:

$$R(\boldsymbol{w}) \equiv R(\boldsymbol{e}, m) = \mathbb{E}_{(\boldsymbol{z}_\star, \boldsymbol{z})} \left[\left\| \boldsymbol{g}(\boldsymbol{z}_\star) - \Pi \left[\operatorname{softmax}\left(\boldsymbol{z}\boldsymbol{z}^\top \right) \right] \right\|_{\mathrm{F}}^2 \right]. \tag{6.9}$$

Note that this formulation reduces the problem of understanding the landscape geometry of R in the $\boldsymbol{w} \in \mathbb{R}^d$ space to understanding it in $(\boldsymbol{e}, m) \in \mathbb{R}^{L+1}$ — a significant simplification when the token size d is large with respect to the sequence length L, the regime we will focus on this work. Note that, given the fixed norm constraint on $\boldsymbol{w}$, the sufficient statistics are constrained inside the unit ball of $\mathbb{R}^{L+1}$: $\left\|(\boldsymbol{e}, m)\right\| \leq 1$.

Escaping mediocrity — As previously discussed, studying the convergence rate of one-pass SGD is akin to studying the population risk landscape. In the standard single-index model, the picture arising from [BGJ21; Arn+24a] is rather simple: the only critical points of the population risk are single global minima at the target weights and (possibly) a strict saddle at zero correlation. Therefore, the convergence rate of one-pass SGD from random initialization is dominated by the time taken to escape this saddle-point, a scenario commonly referred to as *escaping mediocrity* [Arn+24a].

As we shall see, the risk landscape of sequence models is richer, with in particular the presence of local minima. Nevertheless, these models share the common property of mediocrity at initialization, with the convergence rate dominated by the flatness of the initial saddle-point. Therefore, we start our discussion by formalizing this notion in the context of SSI models. In the high-dimensional scenario where d is large, the initial weight $\boldsymbol{w}^0$ is approximately orthogonal to the target direction $\boldsymbol{w}_\star$, as well as the positional embedding P. Quantitatively, the sufficient statistics are distributed as

$$\lim_{d \to +\infty} \sqrt{d}(\boldsymbol{e}^0, m^0) \sim \mathcal{N}(0, I_{L+1}). \tag{6.10}$$

Namely, the initial value of the sufficient statistic is $(\boldsymbol{e}^0, m^0) \approx (\mathbf{0}, 0)$, with fluctuations of order $O(1/\sqrt{d})$. For the model in Equation (6.1), $(\boldsymbol{e}, m) = (\mathbf{0}, 0)$ is a saddle-point of risk, and the dynamics is divided in two phases:

- The escape from the initial condition, where the model develops a weak correlation with the target direction $\boldsymbol{w}_\star$ and/or the positional embedding P;
- Full recovery where it reaches a complete overlap with either the target direction $\boldsymbol{w}_\star$ and/or the positional embedding P: $\left\|(\boldsymbol{e}, m)\right\| \approx 1$.

As previously discussed, the first phase is the one that requires the most number of gradient steps [BGJ21; Arn+24a]: the sample complexity required for the first phase is always greater or equal to the one required for the second phase; after having reached a small correlation with the target, the attention decays exponentially fast to a complete overlapped state.

Definition 6.1 (Weak recovery)**.** *Let $\eta \in (0, 1)$ a parameter independent from d. We say that the model has* weakly recovered *the target when $\left\|(\boldsymbol{e}, m)\right\| \geq \eta$. The weak recovery time is then*

$$\nu_\eta^{\text{weak}} = \min\left\{\nu \geq 0 \colon \left\|(\boldsymbol{e}^\nu, m^\nu)\right\| \geq \eta\right\}.$$

We use this definition of weak recovery as a proxy for identifying the learning has happened, since the subsequent *strong recovery* will be faster. Figure 6.1 shows some examples of population loss surface: apart from initialization, there are no other critical points where the dynamic can get slowed down.

Finally, for simplicity of the discussion we will make the following assumption on the spherical one-pass SGD dynamics in Equation (6.2).

Assumption 6.2 (Gradient flow approximation)**.** *We approximate the training dynamics of Equation* (6.2) *via the following ODE, which corresponds to an order-2 Taylor expansion in γ:*

$$\frac{d\boldsymbol{w}}{dt} = -\mathbb{E}\left[\nabla_{\boldsymbol{w}}^{\perp}\ell(X, y, f_{\boldsymbol{w}})\right] - \frac{\gamma}{2}\mathbb{E}\left[\left\|\nabla_{\boldsymbol{w}}^{\perp}\ell(X, y, f_{\boldsymbol{w}})\right\|^2\right]\boldsymbol{w}, \tag{6.11}$$

where $\nabla_{\boldsymbol{w}}^{\perp} = (I - \boldsymbol{w}\boldsymbol{w}^\top)\nabla_{\boldsymbol{w}}$ is the spherical gradient. The time scaling corresponds to $\nu = t/\gamma$, that is, a continuous time t is reached after t/γ gradient steps.

As shown in [BGJ21; Arn+24c], such an ODE captures both the right weak recovery time for a fixed γ, and the maximal value of γ for which the dynamics do not stay trapped in the uninformative region.

6.2 The sample complexity of SGD

In this section, we focus on understanding the complexity of the SGD algorithm, i.e., how the number of total gradient steps n scales with the dimension d of the token embeddings, in the high-dimensional limit $d \gg 1$; for simplicity of exposition, we focus on the case $k_{\text{out}} = 1$, but the same arguments can be repeated for each of the components of the output.

No positional encoding — We first focus on the case without positional encoding: $P = 0$ implies that the only relevant sufficient statistic is m. An analogy with single-index models for networks can be done: For single-index models, the *information exponent* fully characterizes the sample complexity of the SGD algorithm [BGJ21]. The definition can be generalized for sequential data.

Definition 6.2 (Sequence Information Exponent (SIE))**.** *Given $\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}$ a sequence single-index model, let g be the function that acts on the local field $\boldsymbol{z}_\star$, we define the* sequence information exponent *as*

$$SIE(\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}) \coloneqq \min \left\{ \sum_{l=1}^{L} k_l > 0 \colon \boldsymbol{k} \in \mathbb{N}^L, \mathbb{E}_{\boldsymbol{z}\sim\mathcal{N}(0,I_L)} \left[\left(\prod_{l=1}^{L} \text{He}_{k_l}(z_l) \right) g(\boldsymbol{z}) \right] \neq 0 \right\},$$

where He_k *is the k-th order Hermite polynomial.*

In Appendix F.2 we provide more details on Hermite polynomials. Let us give some examples of SIE for different SSI models:

- $g(\boldsymbol{z}_\star) = z_{\star,1} + z_{\star,2} + \cdots + z_{\star,L}$ has SIE $= 1$;
- $g(\boldsymbol{z}_\star) = z_{\star,1} z_{\star,2} = \text{He}_1(z_{\star,1})\text{He}_1(z_{\star,2})$ has SIE $= 2$;
- $g(\boldsymbol{z}_\star) = \text{He}_1(z_{\star,1})\text{He}_4(z_{\star,2}) + \text{He}_2(z_{\star,3})\text{He}_2(z_{\star,4})$ has SIE $= 4$;
- $g(\boldsymbol{z}_\star) = \prod_{l=1}^{L} \text{He}_{k_l}(z_{\star,l})$ has SIE $= \sum_{l=1}^{L} k_l$.

The main feature of the information exponent is that it can be connected to the sample complexity of the SGD algorithm; we prove an equivalent result for the sequence information exponent.

Here we quote the sharp bound: at SIE ≥ 2 an additional, not-provably-optimal polylogarithmic factor (a $\log d$ at SIE $= 2$) is suppressed. The formal proof in Appendix F.3 carries it in full.

Theorem 6.1 (Informal)**.** *Let $\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}(X)$ be a sequence single-index model, and let SIE be its sequence information exponent. If the model $f_{\boldsymbol{w}}$ has a rich enough Hermite expansion, then the sample complexity of the SGD algorithm is*

$$\nu_\eta^{\text{weak}} = \begin{cases} \mathcal{O}_L(d) & \text{if } SIE = 1 \\ \mathcal{O}_L(d \log d) & \text{if } SIE = 2 \\ \mathcal{O}_L(d^{SIE-1}) & \text{if } SIE \geq 3 \end{cases}.$$

A formal statement of the Theorem and the proof are given in the Appendix F.3. It relies on the following connection between the *flatness* of the landscape near initialization and the sequence information exponent:

Proposition 6.1. *Define the* flatness *index $\kappa(f_{\boldsymbol{w}}, \boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star})$ of the model as*

$$\kappa(f_{\boldsymbol{w}}, \boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}) = \min \left\{ k > 0 \colon \nabla^k R(\boldsymbol{0}, 0) \neq 0 \right\},$$

where R is the reduced population loss defined in Equation (6.9)*. Then, if the model $f_{\boldsymbol{w}}$ has a rich enough Hermite expansion,*

$$\kappa \left(f_{\boldsymbol{w}}, \boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star} \right) = SIE \left(\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star} \right).$$

Intuitively, a higher value of κ implies that the landscape of R is flatter around the initialization point $(\mathbf{0}, 0)$, and thus the number of gradient steps needed to build a weak correlation along either the $\boldsymbol{e}$ or m directions is higher.

In Figure 6.1, we show some examples of the population loss landscape for different values of the SIE. Figure 6.1 focuses on even SIE because the symmetry of Equation (6.1) restricts the possible targets to even functions; in the Appendix F.4 we discuss how to surpass this limitation, and we present settings with odd SIE.

The effect of positional encoding — Despite the fact that positional encoding only acts on the trained model, and not the target function, it changes the population loss, potentially changing the dynamic at initialization. In particular, adding positional encoding increase the expressivity of the model, and can ultimately lead to faster weak-recovery of the SGD.

Lemma 6.1. *Let $f_{\boldsymbol{w}}$ be a model with $P = 0$ that learns a target $\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}(X)$ with a given* SIE*. If we add a positional encoding P to the model, and let $f^{\text{new}}_{\boldsymbol{w}}$ be the new model, then*

$$\kappa\left(f^{\text{new}}_{\boldsymbol{w}}, \boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}\right) \leq \kappa\left(f_{\boldsymbol{w}}, \boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}\right) = \mathit{SIE}\left(\boldsymbol{f}^{\text{SSI}}_{\boldsymbol{w}_\star}\right).$$

In other words, the positional encoding can only decrease the flatness of the loss landscape, thus the sample complexity of the SGD algorithm could be reduced. The proof of Lemma 6.1 is given in Appendix F.3. In the right part of Figure 6.2, we present an example where adding the positional encoding can improve the sample complexity of the SGD algorithm. The left part of Figure 6.2 shows that the population loss landscape at initialization is flat, and the hessian is null: the first non-null term in the Taylor expansion is at order 4, hence the SIE is 4. The right part of Figure 6.2 shows instead that when we add the positional encoding, a non-null term appears at order 2, and $\text{SIE}_{\text{positional}} = 2$. Note that while the positions of the global minima are not affected by the positional encoding, SGD can converge to the new local minima instead; more discussion on this point is given in Section 6.4. In contrast, the bottom row of Figure 6.1 shows that the positional encoding is not necessarily beneficial: there are cases for which the loss landscape changes, but not the sample complexity.

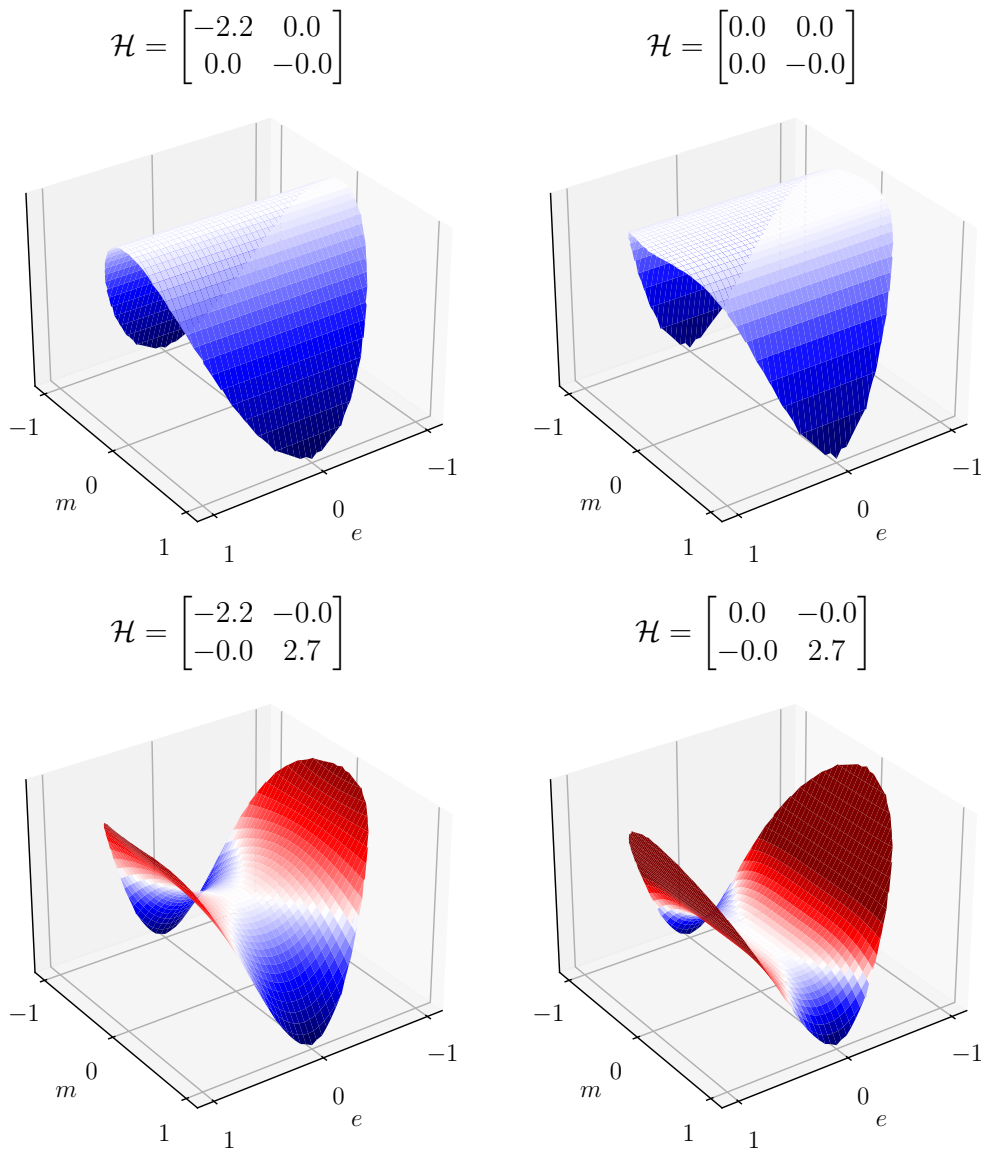


Figure 6.1: The landscape of the population risk, together with the Hessian at initialization, for different values of the SIE and positional encoding. (top-left) $g(\boldsymbol{z}_\star) = \mathrm{He}_2(\boldsymbol{z}_{\star,1}) + \mathrm{He}_2(\boldsymbol{z}_{\star,2})$: SIE=2, no positional encoding: null gradient, but non-null Hessian; (top-right) SIE=4, no positional encoding: the first non-null term at initialization is at the 4th order; (bottom-left) $g(\boldsymbol{z}_\star) = \mathrm{He}_2(z_{\star,1}) + \mathrm{He}_2(z_{\star,2})$: SIE=2, with positional encoding: again dynamic dominated by the Hessian, but we have a positive curvature in the direction of e; (bottom-right) SIE=4, with positional encoding: Hessian is positive semidefinite, and the dynamic is again at 4th order in direction of e. In all the examples $L = 2, P_1 = -P_2, \Pi = \mathrm{Tr}$.

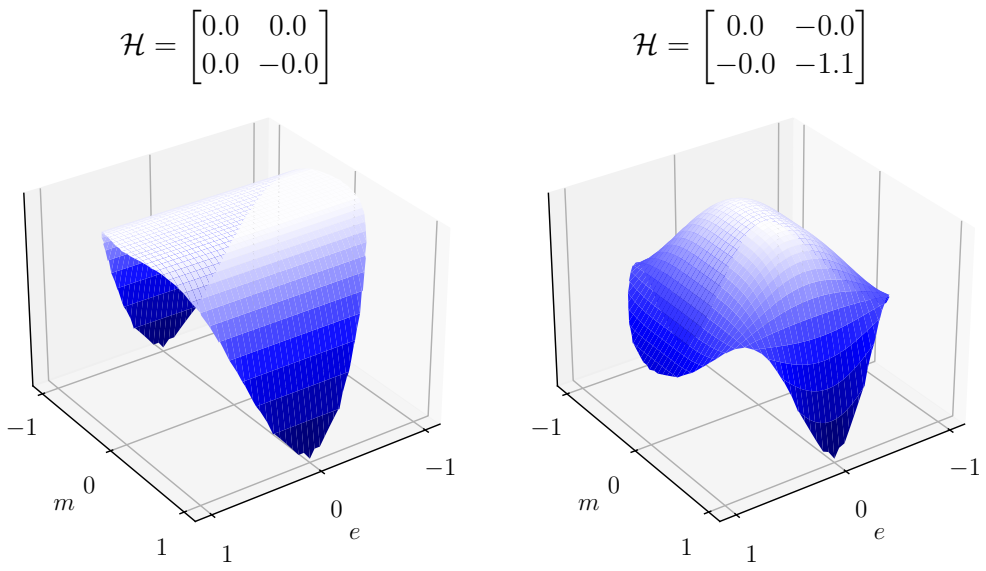


Figure 6.2: Population loss landscape for $P = 0$ (left) and $P \neq 0$ (right). Example of a case where SIE $= 4$, while $\text{SIE}_{\text{positional}} = 2$. Target: $g(\boldsymbol{z}_\star) = 4/3 + \text{He}_4(z_{\star,1}) + 2\text{He}_4(z_{\star,2})$, $P_1 = -P_2$, $\Pi = \text{Tr}$.

6.3 The role of the sequence length

In this section, we focus on a first new emerging characteristic of the *sequence single-index models* over the vanilla *single-index models*: the sequence length L. Therefore, we neglect the effect of the positional encoding, by setting $P = 0$, in order to isolate just the effect of the sequence length. The goal of the section is to measure the speed-up that a model like Equation (6.1) can achieve over the plain *single-index models* when processing sequential data with length L.

Linear attention — Our first goal is to understand the dependence of the convergence rate of SGD on the sequence length. For that, consider the particular case of *linear attention*, given by the reduction map:

$$\Pi[A] = \boldsymbol{a}_{\text{left}}^\top A \boldsymbol{a}_{\text{right}} \quad \text{with } \boldsymbol{a}_{\text{left}} = \boldsymbol{a}_{\text{right}} = \frac{1}{\sqrt{L}} (1, 1, \dots, 1)^\top \in \mathbb{R}^L. \tag{6.12}$$

Rearranging the terms

$$f_{\boldsymbol{w}}(\boldsymbol{z}) = \boldsymbol{a}_{\text{left}}^\top \left(\boldsymbol{z}\boldsymbol{z}^\top \right) \boldsymbol{a}_{\text{right}} = \frac{1}{L} \sum_{i=1}^{L} \sum_{j=1}^{L} z_i z_j = \left(\sum_{i=1}^{L} \frac{z_i}{\sqrt{L}} \right)^2 = \left(\frac{\text{flatten}(X) \cdot \boldsymbol{w}_{\text{tied}}}{\sqrt{L}} \right)^2, \tag{6.13}$$

where $\boldsymbol{w}_{\text{tied}} \coloneqq \text{concat}(\boldsymbol{w}, \dots, \boldsymbol{w}) \in \mathbb{R}^{Ld}$ is the concatenation of L copies of $\boldsymbol{w}$. This model is equivalent to a *generalized linear model* with *tied weights*, and activation function $\sigma(x) = x^2$.

In terms of performance, taking a general activation σ will at worst be the same as the *attention mechanism* originally considered, if not better. More precisely, we consider:

$$f_{\boldsymbol{w}}(X) = \sigma\left(\frac{\text{flatten}(X)\cdot \boldsymbol{w}_{\text{tied}}}{\sqrt{L}}\right). \tag{6.14}$$

This is the most generic model we study in this section. Further numerical experiments elucidating the equivalence of the speed-up for this *tied network* and for the attention models can be found in Appendix F.5. Note that *tied networks* are not restricted to learn even functions only, differently from the model in Equation (6.1).

The corresponding untied network — Given the model in Equation (6.14), a natural benchmark is the model with untied weights. Let $W \in \mathbb{R}^{L\times d}$ be the matrix of weights, whose rows are all updated with Equation (6.2), the untied network is given by

$$f_W(X) = \sigma\left(\frac{\text{flatten}(X)\cdot \text{flatten}(W)}{\sqrt{L}}\right). \tag{6.15}$$

Since we have L independent weights (the rows of W) the sufficient statistic measuring the overlap between the model and the target SSI is not a scalar as for the tied network, but a vector of length L

$$\boldsymbol{m} = \frac{W\boldsymbol{w}_\star}{d} \in \mathbb{R}^L \quad \text{compacted to a scalar as} \quad m_{\text{untied}} = \frac{\|\boldsymbol{m}\|}{\sqrt{L}}. \tag{6.16}$$

Measuring the speed-up — The learning rate plays an important role in determining the number of gradient steps needed to reach weak recovery: the larger the learning rate $\gamma(L)$ is, the faster the model learns. However, if it becomes too large, SGD will fail to converge, never achieving weak recovery. The gradient-flow approximation in Equation (6.11) exhibits this effect: when γ becomes too large the dynamic of the system is not attracted by $\boldsymbol{w}_\star$ or P anymore, and there is no learning. In order to have a faithful measure of the speed-up, we will assume that the learning rate is taken to be the largest possible that guarantees weak recovery; we discuss this upper bound on the learning rate in Appendix F.5.

We measure the speed-up of tied networks with respect to untied networks, as measured in terms of number of gradient steps needed to reach weak recovery, by the ratio of the weak recovery times in the two cases

$$\text{gain}(L) \coloneqq \frac{\nu^{\text{weak}}_{\eta,\text{untied}}}{\nu^{\text{weak}}_{\eta}}, \tag{6.17}$$

where $\nu^{\text{weak}}_{\eta,\text{untied}}$ is given by Definition 6.1 where m is replaced by m_{untied}; the dependence of gain on η is subleading, and we will neglect it in the following.

Theorem 6.2. *Let $C_{\text{SIE}} \in \mathbb{R}^{L^{\text{SIE}}}$ be the first non-zero tensor in the Hermite expansion of g (see Appendix F.2). Then the gain satisfies with high probability*

$$\text{gain} \gtrsim \left(\frac{C_{\text{SIE}}\times(\mathbf{1},\dots,\mathbf{1})}{\|C_{\text{SIE}}\|_{\text{op}}}\right)^2 \cdot \begin{cases} L & \textit{if } \text{SIE}=1 \\ 1 & \textit{otherwise} \end{cases}.$$

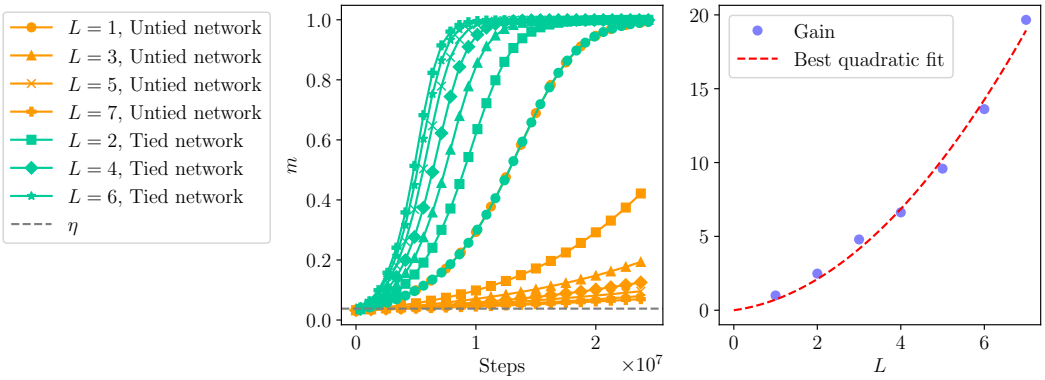


Figure 6.3: Left: overlap m for tied (green) and untied (orange) networks as a function of the number of gradient steps; different symbols represent different values of L. Right: measured gain as a function of the sequence length L, with the best fit line showing its scaling as L^2. $g(\boldsymbol{z}_\star) = \sum_{i=1}^{L} \mathrm{He}_2(z_{\star,i})$, $d = 1000$, $\sigma = \mathrm{ReLU}$.

If the tensor C_{SIE} *is* orthogonally decomposable, *in particular in the cases where* $\mathrm{SIE} \leq 2$ *or* g *is separable, then*

$$\text{gain} \asymp \left(\frac{C_{\mathrm{SIE}} \times (\mathbf{1}, \dots, \mathbf{1})}{\|C_{\mathrm{SIE}}\|_{\mathrm{op}}} \right)^2 \cdot \begin{cases} L & \textit{if } \mathrm{SIE} = 1 \\ 1 & \textit{otherwise} \end{cases}.$$

By definition of the operator norm, we have

$$C_{\mathrm{SIE}} \times (\mathbf{1}, \dots, \mathbf{1}) \leq \|C_{\mathrm{SIE}}\|_{\mathrm{op}} L^{\mathrm{SIE}/2}, \quad \text{and hence} \quad 0 \leq \text{gain} \lesssim L^{\mathrm{SIE} \vee 2}.$$

Since the untied network has L times the number of parameters compared to the tied one, a naive parameter counting argument would yield a gain $= L^{(SIE-1)\vee 1}$ expected gain. Counter-intuitively, the actual gain of using a tied network can either exceed or fall short of this naive value, depending on the function g. In pathological cases (see Appendix F.5), the tied network can even either fail to learn the target, or do so slower than its untied counterpart.

Example for $\mathrm{SIE} = 2$ — Let's assume to have a target function $g(\boldsymbol{z}_\star) = \sum_{i=1}^{L} \mathrm{He}_2(z_{\star,i})$. In this case, the SIE is 2 and the tensor C_2 is simply the identity matrix I_L. The gain is by

$$\text{gain} \asymp \left(\frac{I_L \times (\mathbf{1}, \dots, \mathbf{1})}{\|I_L\|_{\mathrm{op}}} \right)^2 \cdot 1 = \left(\frac{L}{1} \right)^2 \cdot 1 = L^2.$$

Figure 6.3 show a numerical experiment, with a ReLU activation, confirming the result of Theorem 6.2.

6.4 Positional encoding and training dynamics

We now turn our attention to the role played by positional encoding in the attention layer when trained under SGD. Since the focus is on the effect of positional encoding, we stick with a class of

target functions that can exhibit either a *semantic* (label mostly depends on tokens value, but not the order) or a *positional* (where tokens most important feature is their position in the sequence, rather than their embedding). Consider an SSI target function of the form

$$f^{\mathrm{SSI}}_{\boldsymbol{w}_\star}(X) = (1-\omega)\,\mathrm{softmax}\left(X\boldsymbol{w}_\star\boldsymbol{w}_\star^\top X^\top\right) + \omega\,\mathrm{softmax}\left[\begin{pmatrix} a^2 & -a^2 \\ -a^2 & a^2 \end{pmatrix}\right] \in \mathbb{R}^{2\times 2}, \quad (6.18)$$

where the parameter $\omega \in [0,1]$ allows the target to switch from a semantic to a positional behavior, while the parameter $a \in (0,1]$ controls the alignment of the target with its positional part.

We shall train the model in Equation (6.1) with positional encoding $P_1 = -P_2$ and reduction map Π the identity function. We focus on the *gradient flow* limit where η is sufficiently small [Chu54]. Our analysis will focus on the population loss given by Equation (6.9), since it completely characterize the behavior of SGD in this regime; the sufficient statistics $(\boldsymbol{e}, m)$ are the only free variables of the setting.

The sequential information exponent of this setting is SIE $= 2$ (see Appendix F.6 for the explicit derivation), thus by Theorem 6.1 the sample complexity for escaping the initialization is $\mathcal{O}(d\log d)$. After the initial phase, SGD fast converges to the minimum of the population loss that is fully aligned with either the semantic or the positional sufficient statistic, i.e. $\|(\boldsymbol{e}, m)\| = 1$, but is not guaranteed to be the global minimum. Figure 6.5 shows an example where the population loss has two minima, one semantic with $(e, m) = (0, 1)$ and one positional with $(e, m) = (1, 0)$, and the steepest direction at initialization, namely the eigenvector associated with the lowest eigenvalue of the Hessian, points towards the local one; in this case, the gradient flow will converge to the wrong minimum.

Varying the parameters, ω and a the high-dimensional SGD dynamics from random initialization exhibits a rich phenomenology. In Figure 6.4 we show the phase diagram with all the possible behaviors:

- **Unique Positional Minima**: the population loss has unique positional minima, and the SGD converges to it. This is the case for $\omega = 1$ and $a = 1$.
- **Global Positional Minima**: the population loss has both a semantic and a positional minimum, and SGD converges to the global positional one.
- **Global Semantic Minima, Positional Dynamic**: the population loss has both a semantic and a positional minimum, and SGD converges to the local positional one. This is the case where SGD **does not converge** to the global minima.
- **Unique Semantic Minima, Misaligned Dynamics**: the population loss has unique semantic minima, and the SGD converges to it, even though the steepest direction at initialization points orthogonal to it.
- **Global Semantic Minima**: the population loss has both a semantic and positional minima, and SGD converges to the global semantic one.

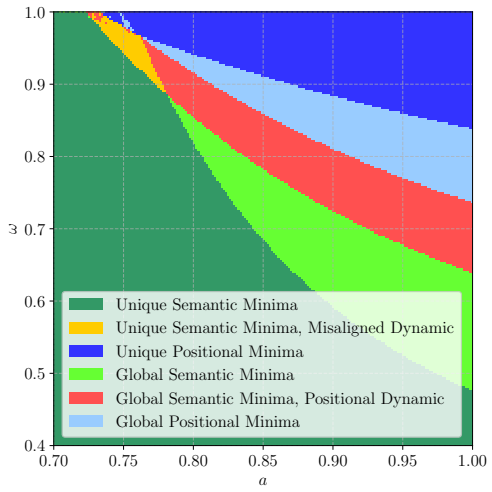


Figure 6.4: Different behaviors of SGD depending on the parameters ω and a.

- **Unique Semantic Minima**: the population loss has unique semantic minima, and the SGD converges to it. This is the case for $\omega = 0$ and $a = 1$.

We verify this by simulating many runs of SGD with different initializations and data samples. In Figure 6.5 we compute the empirical probability of convergence to the semantic minima, for $a = 1$ and varying ω. The theoretical value of ω where we have a transition from sematic to positional dynamics is $\omega_{\text{trans}} = 0.64$, which is in good agreement with the transition observed in the measured probabilities; the transition becomes sharper as d increases: ideally, in the limit $d \to \infty$ we expect a step function. The simulations in Figure 6.5 are performed with $d = 1000$, and some finite size effects are still present. In Appendix F.6 we show present a more detailed analysis, including different values of d.

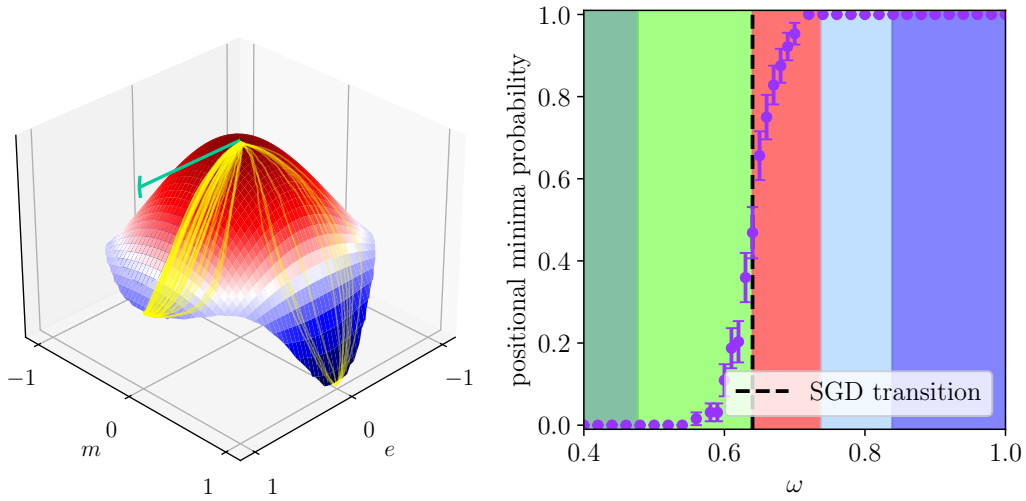


Figure 6.5: (left) surface of the population loss for $\omega = 0.67$ and $a = 1$. The steepest direction at initialization (green vector) points towards the *positional* local minimum, while the global minima is semantic. Some examples of SGD trajectories are shown in yellow: most of them fall into the positional local minimum, while some others manage to fully-recover the global (semantic) minimum due to finite size effects ($d = 1000$). (right) empirical probability of convergence to the semantic minima as a function of ω for $a = 1$. The probability is computed over 64 SGD runs with different initializations and data samples. The theoretical prediction of the transition from semantic to positional minima is at $\omega_{\text{trans}} \approx 0.64$.

6.5 Conclusion

In this paper, we introduced the Sequence-Single Index model as a new, high-dimensional theoretical framework for analyzing single-layer attention architectures. The most significant contribution of this study is the definition of the Sequence Information Exponent. This exponent, which serves as a rigorous tool to quantify the inherent hardness and predict the sample complexity of Stochastic Gradient Descent, is defined in direct analogy with classical single-index models. This analysis transcends qualitative understanding, providing precise scaling laws for the required number of gradient steps, denoted by n, relative to the token dimension, denoted by d. We demonstrated that the sequential setting is significantly richer, showing how the sequence length L accelerates convergence and how positional encoding can proactively reduce the SIE, thereby lowering the computational barrier to learning.

This work establishes a foundational line of research essential for a principled understanding of modern sequential data models, particularly those based on the Transformer architecture. The intricate dynamics manifesting in the population loss landscape, exemplified by the identification of phase transitions and the convergence to suboptimal local minima, unveil a wealth of avenues for future investigation. Subsequent research should aim to fully map these complex high-dimensional dynamics, develop techniques for robustly breaking inherent symmetries, and extend the SIE framework to more complex multi-index and multi-layer attention systems. We hope that this quantitative framework will stimulate further theoretical explorations, leading to the development of a robust, generalizable understanding of learning on structured sequential data.

Limitations — The theoretical analysis of SGD in Sequence Single-Index models is subject to several simplifications for tractability. Specifically, the model under scrutiny is a simplified, single-layer attention architecture where the key and query matrices are tied, the attention head dimension is one ($d_{\text{head}} = 1$), and the value matrix is the identity ($V = I_L$). It should be noted that the scope of this architecture is restricted to a specific class of target functions, including even functions, although Appendix F.4 provides an extension to this class. Additionally, the analysis operates under the assumption of token independence across the sequence, a crucial simplification that facilitates the execution of numerous sequence modeling tasks. The analysis of SGD dynamics is founded on the approximation of the discrete dynamics via a second-order ODE (Gradient Flow approximation).

Acknowledgements

We would like to thank Luca Pesce, Luca Biggio, and Yatin Dandi for their insightful discussions. We acknowledge funding from the Swiss National Science Foundation grants SNSF SMArtNet (grant number 212049), OperaGOST (grant number 200021 200390), DSGIANGO (grant number 225837) and by the French government, managed by the National Research Agency (ANR), under the France 2030 program with the reference "ANR-23-IACL-0008" and the Choose France - CNRS AI Rising Talents program.

Appendix A

Appendices to the Introduction

A.1 The single-index overlap dynamics on the sphere

This appendix supplies the derivation deferred from Section 1.4. Throughout we are in the single-index, single-neuron setting with the read-out fixed at $a = 1$ and the weights confined to the sphere $\|\boldsymbol{w}\| = \sqrt{d}$, so the self-overlap is frozen at $q = 1$ and the pre-activations are jointly Gaussian, $(\lambda, \lambda^\star) \sim \mathcal{N}(0, \Omega)$ with $\Omega = \left(\begin{smallmatrix} 1 & m \\ m & 1 \end{smallmatrix}\right)$. Our goal is the compact overlap equation

$$\frac{\mathrm{d}m}{\mathrm{d}t} = (1 - m^2)\, C'(m), \qquad C(m) = \mathbb{E}_{(\lambda,\lambda^\star)}\big[\sigma(\lambda)\, h^\star(\lambda^\star)\big]\,, \tag{A.1}$$

which we obtain in two complementary ways: directly from the discrete projected updates, in their gradient-flow limit on the sphere, which makes the origin of the factor $(1 - m^2)$ transparent; and by reducing the spherically projected Saad & Solla drift (1.22), which establishes the equivalence claimed in the main text.

A.1.1 Derivation as a spherical gradient flow

We start from the projected update of the main text: at step ν, on a fresh sample $(\boldsymbol{x}, y)$,

$$\boldsymbol{w}^{\nu+1} = \sqrt{d}\,\frac{\boldsymbol{w}^\nu - \gamma\, \nabla_{\boldsymbol{w}} \ell}{\|\boldsymbol{w}^\nu - \gamma\, \nabla_{\boldsymbol{w}} \ell\|}, \qquad \nabla_{\boldsymbol{w}} \ell = \big(\sigma(\lambda) - y\big)\, \sigma'(\lambda)\, \frac{\boldsymbol{x}}{\sqrt{d}}. \tag{A.2}$$

Renormalizing onto the sphere makes the increment tangential to leading order in the step size,

$$\boldsymbol{w}^{\nu+1} - \boldsymbol{w}^\nu = -\gamma\, P^\perp_{\boldsymbol{w}}\, \nabla_{\boldsymbol{w}} \ell + O(\gamma^2), \qquad P^\perp_{\boldsymbol{w}} = I - \frac{\boldsymbol{w}\boldsymbol{w}^\top}{d}, \tag{A.3}$$

the discarded $O(\gamma^2)$ carrying the retraction's curvature together with the gradient fluctuations—the *variance term* Ψ, one power of γ down and neglected throughout, as in the main text. Since the sample is fresh the gradient is unbiased for the population risk, $\mathbb{E}[\nabla_{\boldsymbol{w}} \ell] = \nabla_{\boldsymbol{w}} \mathcal{R}$, and—the projector depending on $\boldsymbol{w}$ alone—

$$\mathbb{E}[\boldsymbol{w}^{\nu+1} - \boldsymbol{w}^\nu] = -\gamma\, P^\perp_{\boldsymbol{w}}\, \nabla_{\boldsymbol{w}} \mathcal{R} + O(\gamma^2). \tag{A.4}$$

On the sphere $q = 1$ is frozen, so $\mathcal{R}$ depends on the weights only through $m = \langle \boldsymbol{w}, \boldsymbol{w}^\star \rangle / d$, and $\mathcal{R}(m) = \text{const} - C(m)$ as in Equation (1.24). The chain rule gives the radial gradient $\nabla_{\boldsymbol{w}} \mathcal{R} = \mathcal{R}'(m)/d\, \boldsymbol{w}^\star$; projecting it leaves $P_{\boldsymbol{w}}^{\perp} \boldsymbol{w}^\star = \boldsymbol{w}^\star - m\, \boldsymbol{w}$, and reading off the overlap increment $\mathbb{E}[\Delta m] = 1/d\, \langle \boldsymbol{w}^\star,\, \mathbb{E}[\Delta \boldsymbol{w}] \rangle$,

$$\mathbb{E}[\Delta m] = -\frac{\gamma}{d^2}\, \mathcal{R}'(m)\, \left(\left\| \boldsymbol{w}^\star \right\|^2 - m\, \langle \boldsymbol{w}^\star, \boldsymbol{w} \rangle \right) = -\frac{\gamma}{d}\, \mathcal{R}'(m)\, (1 - m^2), \tag{A.5}$$

using $\|\boldsymbol{w}^\star\|^2 = d$ and $\langle \boldsymbol{w}^\star, \boldsymbol{w} \rangle = d\, m$. The Saad & Solla time scaling now makes the dimension factor meaningful: measuring time as $t = \nu\, \gamma / d$, so that one sample advances t by $\delta t = \gamma/d$, the overlap concentrates in the high-dimensional limit on the deterministic trajectory $\dfrac{\mathrm{d}m}{\mathrm{d}t} = \mathbb{E}[\Delta m]/\delta t$. With $\mathcal{R}'(m) = -C'(m)$ this is

In the single-index setting the time scaling is $t = \nu\, \gamma/d$ because $p = 1$.

$$\frac{\mathrm{d}m}{\mathrm{d}t} = -(1 - m^2)\, \mathcal{R}'(m) = (1 - m^2)\, C'(m), \tag{A.6}$$

Equation (A.1). The factor γ/d is thus the time step itself—not a stray constant—and $(1 - m^2)$ is the spherical Jacobian $\sin^2 \theta$ for $m = \cos\theta$.

A.1.2 Equivalence with the Saad & Solla equation

The spherically projected Saad & Solla drift (1.22) reads

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \mathbb{E}\big[(h^\star(\lambda^\star) - \sigma(\lambda))\, \sigma'(\lambda)\, \lambda^\star \big] - m\, \mathbb{E}\big[(h^\star(\lambda^\star) - \sigma(\lambda))\, \sigma'(\lambda)\, \lambda \big]\,. \tag{A.7}$$

For the bivariate Gaussian above, Stein's identities read, for any regular $G(\lambda, \lambda^\star)$,

$$\mathbb{E}[\lambda\, G] = \mathbb{E}[\partial_\lambda G] + m\, \mathbb{E}[\partial_{\lambda^\star} G], \qquad \mathbb{E}[\lambda^\star G] = m\, \mathbb{E}[\partial_\lambda G] + \mathbb{E}[\partial_{\lambda^\star} G], \tag{A.8}$$

since $\mathrm{Var}\, \lambda = \mathrm{Var}\, \lambda^\star = 1$ and $\mathrm{Cov}(\lambda, \lambda^\star) = m$. Applying them to $G = (h^\star(\lambda^\star) - \sigma(\lambda))\, \sigma'(\lambda)$, whose partial derivatives are $\partial_\lambda G = (h^\star - \sigma)\, \sigma'' - (\sigma')^2$ and $\partial_{\lambda^\star} G = h^{\star\prime}\, \sigma'$ (arguments suppressed), the two terms of the drift become

$$\mathbb{E}[\lambda^\star G] = m\, \mathbb{E}\Big[(h^\star - \sigma)\sigma'' - (\sigma')^2 \Big] + \mathbb{E}\Big[h^{\star\prime} \sigma' \Big]\,, \tag{A.9}$$

$$m\, \mathbb{E}[\lambda\, G] = m\, \mathbb{E}\Big[(h^\star - \sigma)\sigma'' - (\sigma')^2 \Big] + m^2\, \mathbb{E}\Big[h^{\star\prime} \sigma' \Big]\,. \tag{A.10}$$

Subtracting, the self-interaction terms cancel identically and only the cross term survives,

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \mathbb{E}[\lambda^\star G] - m\, \mathbb{E}[\lambda\, G] = (1 - m^2)\, \mathbb{E}\Big[h^{\star\prime}(\lambda^\star)\, \sigma'(\lambda) \Big]\,. \tag{A.11}$$

Price's theorem—which states that, for jointly Gaussian $(\lambda, \lambda^\star)$ of unit variance and correlation m,

$$C'(m) = \frac{\mathrm{d}}{\mathrm{d}m}\, \mathbb{E}[\sigma(\lambda)\, h^\star(\lambda^\star)] = \mathbb{E}[\sigma'(\lambda)\, h^{\star\prime}(\lambda^\star)] \tag{A.12}$$

—then identifies this expectation with $C'(m)$ and recovers Equation (A.1), establishing the equivalence.

A.2 Validity window of the deterministic description

The order-parameter ODEs of Equation (SS-ODEs) rest on a non-asymptotic guarantee, Theorem 1.1: over any fixed horizon $[0, T]$ the stochastic overlap process stays within $C(T)/\sqrt{d}$ of the deterministic trajectory. The horizon enters only through the constant $C(T)$, and it is worth making its growth explicit. Collecting the order parameters into a single vector Ω, the sharper form established in Chapter 2 (Theorem 2.1), due to Veiga et al. [Vei+22], reads, with step size $\delta t = \gamma/pd$ and continuous time $t = \nu\,\delta t$,

$$\|\Omega^{\nu} - \Omega(t)\|_{\infty} \;\le\; e^{Ct}\sqrt{\frac{\gamma}{pd}}, \qquad C > 0. \tag{A.13}$$

Since $\gamma/p = \mathcal{O}(1)$ we have $\sqrt{\gamma/pd} \asymp d^{-1/2}$, so (A.13) is the bound $C(T)/\sqrt{d}$ with the horizon dependence made visible: $C(T) \asymp e^{CT}$, the guarantee degrading *exponentially* in time. It is this bound that we invoked when we called the deterministic description "valid beyond the high-dimensional limit" in Section 1.3.5: the right-hand side vanishes whenever $\gamma/pd \to 0^{+}$, however that limit is reached. What that statement leaves implicit, and what we make precise here, is that the exponential prefactor confines the guarantee to a finite time horizon.

The horizon. Fix a tolerance $\eta \in (0, 1)$ below which we regard the deterministic trajectory as tracking the true process. The right-hand side of (A.13) stays under η exactly while

$$e^{Ct}\sqrt{\frac{\gamma}{pd}} \le \eta \quad\Longleftrightarrow\quad t \;\le\; t_{\star} \coloneqq \frac{1}{2C}\log\frac{pd}{\gamma} - \frac{1}{C}\log\frac{1}{\eta} \;=\; \frac{1}{2C}\log d + \mathcal{O}(1), \tag{A.14}$$

again using $\gamma/p = \mathcal{O}(1)$. The deterministic ODE is therefore controlled only up to $t_{\star} = \mathcal{O}(\log d)$; past it, the error bar (A.13) exceeds any fixed threshold and the description guarantees nothing.

Where this leaves the escape times. Comparing $t_{\star}$ with the information-exponent escape times of Equation (1.32) shows precisely which part of the dynamics the deterministic ODE captures:

- $\ell = 1$: $t_{\text{escape}} \asymp 1 \ll t_{\star}$; the escape happens well inside the window and the deterministic description is exact throughout.
- $\ell = 2$: $t_{\text{escape}} \asymp \log d$, the *same* order as $t_{\star} \asymp {}^{1}/_{2C}\log d$. The escape sits at the very edge of the guarantee: the deterministic drift still fixes the correct scaling, but whether the trajectory is resolved all the way out depends on the constant C.
- $\ell \ge 3$: $t_{\text{escape}} \asymp d^{\ell/2-1}$, polynomially larger than $t_{\star}$. Here $e^{Ct_{\text{escape}}} = e^{C\,d^{\ell/2-1}}$ is super-polynomially large long before escape, and the bound (A.13) is *vacuous* over the entire search phase.

There is a complementary way to see the failure that does not even appeal to the horizon. Near the mediocrity saddle the overlap starts at $m^0 \asymp d^{-1/2}$, while the additive error floor in (A.13) is already $\sqrt{\gamma/pd} \asymp d^{-1/2}$ at $t = 0$. The uncertainty on m is thus of the same order as m itself throughout the flat region. Equivalently, the bound controls a *relative* error on m only

once $m = \mathcal{O}(1)$; while $m = \mathcal{O}(d^{-1/2})$ the signal and the error bar coincide, so the guarantee is empty exactly where the escape is decided. The deterministic mean therefore cannot resolve the sub-$d^{-1/2}$ growth that carries the student out of the saddle—the exponential prefactor only compounds this—and Theorem 1.1 no longer guarantees any prediction across the search phase. This is the analytic content of the statement, made in Section 1.4, that in the search phase the fluctuations of the order parameter cannot be neglected.

What replaces the deterministic argument. The deterministic ODE is therefore used in this thesis only where the horizon is $\mathcal{O}(1)$ or $\mathcal{O}(\log d)$ — the descent phase, and the gradient-flow / mean-field / high-dimensional regimes of the phase diagram in Figure 1.3. The long search phase, which for $\ell \geq 3$ is the bottleneck that sets the sample complexity, is instead controlled by *separate stochastic arguments*, and Chapter 3 is where this is worked out in full and checked against simulation. There, the subsection *"Does stochasticity matter?"* (Section 3.3.3) promotes the overlap ODE to the effective diffusion of Ben Arous, Gheissari, and Jagannath [BGJ22] — same drift, fluctuation amplitude $\sqrt{\gamma/d}$ — and integrates it *through* the saddle, exactly where the bound (A.13) is vacuous. Solving the stochastic exit-time problem yields corrections of relative order $\sqrt{\gamma/d}$ that leave the deterministic exit-time formulas (3.22)–(3.23) unchanged at leading order; and Figure 3.1 confirms numerically, at $d = 3000$, that SGD and the integrated SDE started from the same initialization reach the signal at essentially the same time. Stochasticity is thus *shown*, not merely assumed, not to alter the sample complexity.

The same conclusion is reached, by different means, in the chapters that need polynomially long horizons directly: Chapters 5 and 4 control the discrete process with high probability on the random horizon $t \leq T \wedge \tau_\eta$, through a martingale decomposition of the drift, a discrete Grönwall inequality, and sub-Gaussian concentration of the per-step fluctuations — none of which passes through (A.13). All are instances of the information-exponent analysis of Ben Arous, Gheissari, and Jagannath [BGJ21]: the deterministic reduction makes the *descent* phase tractable, while the *search* phase belongs to the stochastic theory.

A.3 The partial-trace algorithm

This appendix describes the spectral algorithm of Anandkumar et al. [Ana+17], in the form used by Damian et al. [Dam+24], that achieves weak recovery of the teacher direction $\boldsymbol{w}^\star$ from $n \asymp d^{\ell/2}$ samples, matching the CSQ lower bound discussed in Section 1.4. We work in the single-index setting: $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$, $\boldsymbol{w}^\star \in \mathbb{S}^{d-1}(\sqrt{d})$, $\lambda^\star = \langle \boldsymbol{w}^\star, \boldsymbol{x} \rangle / \sqrt{d} \sim \mathcal{N}(0,1)$, and $y = h^\star(\lambda^\star) = \sum_{k \geq \ell} a_k \operatorname{He}_k(\lambda^\star)$ with information exponent ℓ and $a_\ell \neq 0$. Let $\hat{\boldsymbol{w}}^\star = \boldsymbol{w}^\star / \sqrt{d}$ denote the unit teacher direction.

A.3.1 The population identity

The algorithm rests on the family of multivariate Hermite tensors. For $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$ and $\ell \geq 1$, the order-ℓ Hermite tensor $T_\ell(\boldsymbol{x}) \in (\mathbb{R}^d)^{\otimes \ell}$ is the unique degree-ℓ symmetric tensor polynomial in $\boldsymbol{x}$ orthogonal to all polynomials of lower degree in $L^2(\mathcal{N}(0, I_d))$. The explicit component

forms for small ℓ are:

$$\begin{aligned}
T_1(\boldsymbol{x})_i &= x_i,\\
T_2(\boldsymbol{x})_{ij} &= x_i x_j - \delta_{ij},\\
T_3(\boldsymbol{x})_{ijk} &= x_i x_j x_k - \big(\delta_{ij}\, x_k + \delta_{ik}\, x_j + \delta_{jk}\, x_i\big),\\
T_4(\boldsymbol{x})_{ijkl} &= x_i x_j x_k x_l\\
&\quad - \big(\delta_{ij}\, x_k x_l + \delta_{ik}\, x_j x_l + \delta_{il}\, x_j x_k + \delta_{jk}\, x_i x_l + \delta_{jl}\, x_i x_k + \delta_{kl}\, x_i x_j\big)\\
&\quad + \big(\delta_{ij}\delta_{kl} + \delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\big).
\end{aligned}$$

Each formula is the outer product $\boldsymbol{x}^{\otimes \ell}$ with the lower order corrections needed to enforce orthogonality: at order $\ell = 2$ the identity is subtracted; at order $\ell = 3$, the three symmetrized terms $\delta_{ij}x_k + \delta_{ik}x_j + \delta_{jk}x_i$; at order $\ell = 4$, six rank-two matrix terms and three rank-zero (double-δ) corrections. The general construction follows the recurrence

$$T_\ell(\boldsymbol{x}) = \mathrm{sym}\big(\boldsymbol{x} \otimes T_{\ell-1}(\boldsymbol{x})\big) - (\ell-1)\ \mathrm{sym}\big(I \otimes T_{\ell-2}(\boldsymbol{x})\big), \quad T_0 = 1,\ T_1(\boldsymbol{x}) = \boldsymbol{x}, \tag{A.15}$$

where sym denotes full symmetrization over all index permutations. This mirrors the scalar recurrence $\mathrm{He}_\ell(t) = t\,\mathrm{He}_{\ell-1}(t) - (\ell-1)\,\mathrm{He}_{\ell-2}(t)$.

The defining property, which one verifies from (A.15) by induction, is the contraction identity: for any unit vector $\boldsymbol{v} \in \mathbb{S}^{d-1}$,

$$T_\ell(\boldsymbol{x})[\underbrace{\boldsymbol{v}, \ldots, \boldsymbol{v}}_{\ell}] = \mathrm{He}_\ell(\langle \boldsymbol{v}, \boldsymbol{x}\rangle). \tag{A.16}$$

For example, when $\ell = 3$ this is immediate: contracting with $\boldsymbol{v}^{\otimes 3}$,

$$T_3(\boldsymbol{x})[\boldsymbol{v},\boldsymbol{v},\boldsymbol{v}] = (\langle \boldsymbol{v}, \boldsymbol{x}\rangle)^3 - \underbrace{\big(\delta_{ij}x_k + \delta_{ik}x_j + \delta_{jk}x_i\big)v_i v_j v_k}_{= 3\,\|\boldsymbol{v}\|^2\langle \boldsymbol{v},\boldsymbol{x}\rangle\, =\, 3\langle \boldsymbol{v},\boldsymbol{x}\rangle} = \mathrm{He}_3(\langle \boldsymbol{v}, \boldsymbol{x}\rangle),$$

since each term $\delta_{ij}x_k v_i v_j v_k = \|\boldsymbol{v}\|^2\langle \boldsymbol{v}, \boldsymbol{x}\rangle = \langle \boldsymbol{v}, \boldsymbol{x}\rangle$.

Setting $\boldsymbol{v} = \hat{\boldsymbol{w}}^\star$ in (A.16) returns $\mathrm{He}_\ell(\lambda^\star)$. The Hermite expansion $h^\star(\lambda^\star) = \sum_{k\ge\ell} a_k\,\mathrm{He}_k(\lambda^\star)$ therefore rewrites as a series of fully contracted tensors:

$$h^\star(\lambda^\star) = \sum_{k\ge\ell} a_k\, T_k(\boldsymbol{x})\big[\underbrace{\hat{\boldsymbol{w}}^\star, \ldots, \hat{\boldsymbol{w}}^\star}_{k}\big]. \tag{A.17}$$

The *population identity* follows from tensor Hermite orthogonality. For each k, taking the expectation of (A.17) multiplied by $T_\ell(\boldsymbol{x})$:

$$\mathbb{E}\Big[T_k(\boldsymbol{x})\big[(\hat{\boldsymbol{w}}^\star)^{\otimes k}\big]\cdot T_\ell(\boldsymbol{x})\Big] = \mathbb{E}\big[\mathrm{He}_k(\lambda^\star)\cdot T_\ell(\boldsymbol{x})\big]\,.$$

For $k \neq \ell$ this vanishes because tensors of different degree are orthogonal in $L^2(\mathcal{N}(0, I_d))$. For $k = \ell$, the second-moment identity $\mathbb{E}[(T_\ell(\boldsymbol{x}))_{i_1\cdots i_\ell}(T_\ell(\boldsymbol{x}))_{j_1\cdots j_\ell}] = \ell!\,\delta^{\mathrm{sym}}_{(i_1\cdots i_\ell)(j_1\cdots j_\ell)}$, with δ^{sym} the symmetrized Kronecker product, gives after contracting ℓ indices against $\hat{\boldsymbol{w}}^\star$:

$$\mathbb{E}\big[\mathrm{He}_\ell(\lambda^\star)\cdot T_\ell(\boldsymbol{x})_{j_1\cdots j_\ell}\big] = \ell!\,\hat{w}^\star_{j_1}\cdots\hat{w}^\star_{j_\ell}.$$

Summing over k, and using $a_k = 0$ for $k < \ell$ by definition of the information exponent, the series collapses to the single term $k = \ell$:

$$\mathbb{E}\big[\, y \cdot T_\ell(\boldsymbol{x}) \,\big] = a_\ell \cdot \ell! \cdot (\hat{\boldsymbol{w}}^\star)^{\otimes \ell}. \tag{A.18}$$

The right-hand side is a rank-one spike of $\Theta(1)$ magnitude pointing exactly in the teacher direction. Observing this tensor directly would achieve recovery, but it has d^ℓ entries and we have only n samples. The partial trace collapses it to a matrix or a vector while preserving the rank-one structure.

A.3.2 The estimator

Form the empirical version of Equation (A.18) and contract identity pairs to reduce the order of the tensor. Write $[\, I^{\otimes r} \,]$ for the operation of contracting r pairs of free indices against the $d \times d$ identity.

ℓ **even.** Contract $(\ell - 2)/2$ index pairs, landing on a $d \times d$ matrix:

$$M_n = \frac{1}{n} \sum_{i=1}^{n} y_i \cdot T_\ell(\boldsymbol{x}_i) \Big[\, I^{\otimes (\ell-2)/2} \,\Big] \;\in\; \mathbb{R}^{d \times d}, \qquad \hat{\boldsymbol{w}} = v_1(M_n), \tag{A.19}$$

where $v_1(\cdot)$ denotes the leading eigenvector.

ℓ **odd.** Contract $(\ell - 1)/2$ index pairs, landing on a vector:

$$\boldsymbol{v}_n = \frac{1}{n} \sum_{i=1}^{n} y_i \cdot T_\ell(\boldsymbol{x}_i) \Big[\, I^{\otimes (\ell-1)/2} \,\Big] \;\in\; \mathbb{R}^{d}, \qquad \hat{\boldsymbol{w}} = \frac{\boldsymbol{v}_n}{\|\boldsymbol{v}_n\|}. \tag{A.20}$$

Taking the expectation in (A.19) and using (A.18): each traced identity pair uses $\langle \hat{\boldsymbol{w}}^\star, \hat{\boldsymbol{w}}^\star \rangle = 1$, so contracting $(\hat{\boldsymbol{w}}^\star)^{\otimes \ell}$ against $(\ell - 2)/2$ identity matrices leaves a rank-one matrix:

$$\mathbb{E}[M_n] = a_\ell \cdot \ell! \cdot \hat{\boldsymbol{w}}^\star (\hat{\boldsymbol{w}}^\star)^\top = \frac{a_\ell \cdot \ell!}{d} \cdot \boldsymbol{w}^\star (\boldsymbol{w}^\star)^\top, \tag{A.21}$$

a rank-one spike of strength $a_\ell \cdot \ell! = \Theta(1)$ along $\boldsymbol{w}^\star$. In the odd case, $\mathbb{E}[\boldsymbol{v}_n] = a_\ell \cdot \ell! \cdot \hat{\boldsymbol{w}}^\star$, a vector of $\Theta(1)$ norm aligned with $\boldsymbol{w}^\star$.

A.3.3 Why $n \asymp d^{\ell/2}$ is the threshold

Consider the even case; the odd case is analogous. Decompose $M_n = \mathbb{E}[M_n] + (M_n - \mathbb{E}[M_n])$.

Signal. From (A.21), the spike has operator norm $\big\|\mathbb{E}[M_n]\big\|_{\text{op}} = \Theta(1)$.

Noise. The fluctuation $M_n - \mathbb{E}[M_n]$ is an average of n i.i.d. mean zero matrices. Each summand $y_i \cdot T_\ell(\boldsymbol{x}_i)[\, I^{\otimes (\ell-2)/2} \,]$ has $O(1)$ entries, but a large operator norm: contracting the order-ℓ Hermite tensor down to a matrix leaves a residual operator norm growth of $d^{(\ell-2)/2}$, which combined

with the matrix dimension d places the fluctuation scale at

$$\big\|M_n - \mathbb{E}[M_n]\big\|_{\mathrm{op}} \asymp \sqrt{\frac{d^{\ell/2}}{n}} \cdot \sqrt{\mathbb{E}[h^{\star 2}]}.$$

This is the heavy-tailed noise phenomenon: unlike standard PCA, where the contracted noise has operator norm $d^{1/2}$, the contracted degree-ℓ Hermite tensor yields operator norm $d^{\ell/2}$.

Detectability. The spike is visible above the bulk if and only if signal $\gtrsim$ noise:

$$\Theta(1) \gtrsim \sqrt{\frac{d^{\ell/2}}{n}} \qquad \Longleftrightarrow \qquad \boxed{n \gtrsim d^{\ell/2}.} \tag{A.22}$$

Above this threshold the leading eigenvector of M_n satisfies $\langle \hat{\boldsymbol{w}}, \boldsymbol{w}^\star \rangle \geq \text{const} > 0$, achieving weak recovery. Below it the spike is submerged in the semicircular bulk (a BBP-type transition at $n \asymp d^{\ell/2}$) and $|\hat{\boldsymbol{w}} \cdot \boldsymbol{w}^\star| \approx d^{-1/2}$, no better than a random initialization.

A.3.4 Familiar special cases

$\ell = 2$: The contraction $[\, I^{\otimes 0} \,]$ is trivial, and $T_2(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{x}^\top - I$, so the estimator reduces to the standard PCA matrix,

$$M_n = \frac{1}{n} \sum_{i=1}^{n} y_i \Big(\boldsymbol{x}_i \boldsymbol{x}_i^\top - I \Big), \qquad \mathbb{E}[M_n] = 2\, a_2\, \hat{\boldsymbol{w}}^\star (\hat{\boldsymbol{w}}^\star)^\top,$$

recovering $\boldsymbol{w}^\star$ at $n \asymp d$.

$\ell = 3$: The vector estimator $\boldsymbol{v}_n = \frac{1}{n} \sum_i y_i\, T_3(\boldsymbol{x}_i)[\, I \,]$ contracts one index pair of the third-order tensor to a vector aligned with $\hat{\boldsymbol{w}}^\star$, recovering $\boldsymbol{w}^\star$ at $n \asymp d^{3/2}$.

Appendix B

Appendices to "From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layers networks"

B.1 Effect of noise on Equations (2.14)

In this appendix we present some corrective terms to Equations (2.14) that allows to have better results when numerically integrating ODEs and comparing them to simulations.

In [Vei+22] the terms proportional to γ/p are neglected completely when running numerical integrations. However, we noticed that a term from $\Psi_{ij}^{(\mathrm{Var})}(\Omega)$ could be kept in order to get better agreement between ODEs and simulation in presence of label noise at small but finite $\gamma/p \ll 1$. Letting $\mathcal{E}^* \coloneqq \mathcal{E} - \sqrt{\Delta}z$, we can decompose

$$\Psi_{ij}^{(\mathrm{Var})}(\Omega) = \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_i)\sigma'(\lambda_j)\mathcal{E}^{*2}\right] + \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_i)\sigma'(\lambda_j)\Delta\right],$$

allowing us to define

$$\Psi_{ij}^{(\mathrm{noise})}(\Omega) \coloneqq \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_i)\sigma'(\lambda_j)\Delta\right]. \tag{B.1}$$

The asymptotic of this term at late times $t \gg 1$ was explicitly computed in [BS95] and [Gol+19] for different architectures, where it was found that $\Psi_{ij}^{(\mathrm{noise})} \propto \Delta$. In these cases, when the dynamics approaches the point where $\mathcal{E}^* \sim \gamma\Delta/p$, then the term $\Psi_{ij}^{(\mathrm{GF})}(\Omega)$ is of the same order of $\gamma/p\Psi_{ij}^{(\mathrm{noise})}(\Omega)$, while the remaining part of $\Psi_{ij}^{(\mathrm{Var})}(\Omega)$ is still negligible since is scaling as $\mathcal{E}^{*2}$. It

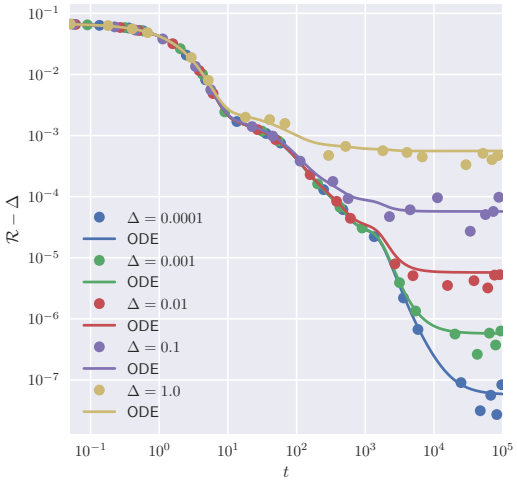


Figure B.1: comparison between the simulated learning dynamic and the one obtained by integrating the differential equations, in the *classical regime*; $k = 4, p = 8, \gamma = 5 \times 10^{-2}, d = 10, \sigma = \mathrm{erf}(\cdot/\sqrt{2})$. Learning the same teacher with different level of noise.

follows that the first order correction in γ/p of equations (2.14) is given by:

$$\frac{\mathrm{dM}}{\mathrm{d}t} = \Psi^{(\mathrm{M})}(\Omega), \qquad \frac{\mathrm{dQ}}{\mathrm{d}t} = \Psi^{(\mathrm{GF})}(\Omega) + \frac{\gamma}{p}\Psi^{(\mathrm{noise})}(\Omega). \qquad \text{(GF-ODE-NOISE)}$$

Despite vanishing in the true $\gamma/p \to 0^+$ limit, the new term enables ODE to catch the behaviour for large times, when simulating small but finite γ/p.

Figure B.1 shows a numerical experiment explicitly designed to show the effect of noise term. The fluctuations visible in the final plateaus result from the stochastic process in Equation (2.9): $\Psi^{(\mathrm{Var})}$ and consequently $\Psi^{(\mathrm{noise})}$ are proportional to $||\boldsymbol{x}^\nu||_2^2$. When $d = O(1)$ then this term is not concentrating to 1 and leads to a fluctuation. Let us again stress the fact that this is an effect visible only when performing numerical experiments with finite GP, whereas in the true limit the fluctuations disappear and the dynamic is described by deterministic ODEs.

B.2 Derivation of Equation (2.21) and local fields covariance

Let's start by taking the time derivative of Equation (2.20)

$$\frac{\mathrm{dQ}^\perp}{\mathrm{d}t} = \frac{\mathrm{dQ}}{\mathrm{d}t} - \frac{\mathrm{dM}}{\mathrm{d}t}\mathrm{P}^{-1}\mathrm{M}^\top - \mathrm{M}\mathrm{P}^{-1}\frac{\mathrm{dM}^\top}{\mathrm{d}t} := \Psi^\perp(\Omega),$$

and plugging in equations (2.14) and (2.12) one after the other we get

$$\Psi^{\perp}_{ij}(\Omega) = \mathbb{E}\Bigg[\left(\sigma'(\lambda_i)\lambda_j + \sigma'(\lambda_j)\lambda_i\right)\mathcal{E} - \sum_{r=1}^{k}\sum_{t=1}^{k}\sigma'(\lambda_i)\lambda^{\star}_r\mathcal{E}\left[\mathrm{P}^{-1}\right]_{rt}\left[\mathrm{M}^{\top}\right]_{tj} - \\ - \sum_{r=1}^{k}\sum_{t=1}^{k}\left[\mathrm{M}\right]_{ir}\left[\mathrm{P}^{-1}\right]_{rt}\lambda^{\star}_t\sigma'(\lambda_i)\mathcal{E}\Bigg] \\ = \mathbb{E}\left[\mathcal{E}\sum_{r=1}^{k}\sum_{t=1}^{k}\left(\sigma'(\lambda_i)\left(\lambda_j - \lambda^{\star}_r\left[\mathrm{P}^{-1}\right]_{rt}\left[\mathrm{M}^{\top}\right]_{tj}\right) + \sigma'(\lambda_j)\left(\lambda_i - \left[\mathrm{M}\right]_{ir}\left[\mathrm{P}^{-1}\right]_{rt}\lambda^{\star}_t\right)\right)\right],$$

where all the expected value are intended over $(\boldsymbol{\lambda}, \boldsymbol{\lambda}^{\star}) \sim \mathcal{N}(\mathbf{0}_{p+k}, \Omega)$. Starting from the definition of $\boldsymbol{\lambda}^{\perp}$

$$\boldsymbol{\lambda}^{\perp} = W^{\perp}\boldsymbol{x} = \left(W - MP^{-1}W^{\star}\right)\boldsymbol{x} = \boldsymbol{\lambda} - MP^{-1}\boldsymbol{\lambda}^{\star},$$

and writing single component

$$\boldsymbol{\lambda}^{\perp}_i = \lambda_i - \sum_{r=1}^{k}\sum_{t=1}^{k}\left[\mathrm{M}\right]_{ir}\left[\mathrm{P}^{-1}\right]_{rt}\lambda^{\star}_t,$$

substituting in the expression above we finally get

$$\Psi^{\perp}_{ij}(\Omega) = \mathbb{E}\left[\left(\sigma'(\lambda_i)\lambda^{\perp}_j + \sigma'(\lambda_j)\lambda^{\perp}_i\right)\mathcal{E}\right]. \tag{B.2}$$

The explicit computation of the expected value depends on the particular activation function used. Even though, the final expression can only be function of the covariance matrix entries, since all the local fields are zero-mean Gaussian variables. We report the covariance matrix of local fields

$$\mathrm{Cov}\left[\boldsymbol{\lambda}, \boldsymbol{\lambda}^{\perp}, \boldsymbol{\lambda}^{\star}\right] = \begin{pmatrix} \mathrm{Q} & \mathrm{Q}^{\perp} & \mathrm{M} \\ \mathrm{Q}^{\perp} & \mathrm{Q}^{\perp} & 0 \\ \mathrm{M}^{\top} & 0 & \mathrm{P} \end{pmatrix}, \tag{B.3}$$

where $\mathrm{Q}^{\perp} = \mathrm{Q} - \mathrm{M}\mathrm{P}^{-1}\mathrm{M}^{\top}$ as defined above.

B.3 Mean-field approximation: proof of Theorem 2.2

Preliminaries We begin by recalling the results of [MMM19]. For any distribution μ on $\mathbb{R}^d$, we define the network function

$$\hat{f}_{\mu}(\boldsymbol{x}) = \int \sigma(\boldsymbol{w}^{\top}\boldsymbol{x})d\mu(\boldsymbol{w}). \tag{B.4}$$

It is easy to check that when

$$\mu = \mu_p(W) := \frac{1}{p}\sum_{i=1}^{p}\delta_{\boldsymbol{w}_i},$$

then $\hat{f}_\mu = f_\Theta$, where f_Θ was defined in (2.1). The associated network risk is then given by

$$\hat{\mathcal{R}}(\mu) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(\mathbf{0},{}^1\!/_{dI_d})}\left[\frac{1}{2}\left(f_{\Theta^\star}(\boldsymbol{x}) - \hat{f}_\mu(\boldsymbol{x})\right)^2\right]. \tag{B.5}$$

Similarly, we can consider the continuous equivalent of the gradient flow equation as follows:

$$\varphi(\boldsymbol{w},\mu) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(\mathbf{0},{}^1\!/_{dI_d})}\left[\sigma'(\boldsymbol{w}^\top\boldsymbol{x})\boldsymbol{x}\left(f_{\Theta^\star}(\boldsymbol{x}) - \hat{f}_\mu(\boldsymbol{x})\right)\right]. \tag{B.6}$$

Through conservation of matter arguments, the evolution of the empirical measure for particles following the gradient flow equation (2.15) obeys the following partial differential equation:

$$\partial_t\mu_t = \nabla_{\boldsymbol{w}}\cdot\big(\mu_t\varphi(\cdot,\mu_t)\big) \tag{GF-PDE}$$

Theorem B.1 ([MMM19], Propositions 13-16)**.** *Let μ_0 be a K^2/d-subgaussian distribution, and consider the following two processes:*

- *the solution μ_t of* (GF-PDE) *with initial value μ_0,*
- *the solution $W(t)$ of the gradient flow ODE* (2.15)*, with initial conditions $\boldsymbol{w_i}(0) \sim \mu_0$ i.i.d.*

Then for any $T \geq 0$ we have

$$\sup_{t\in[0,T]}\left|\mathcal{R}(W(t)) - \hat{\mathcal{R}}(\mu_t)\right| \leq \frac{Ce^{CT}\left(\sqrt{\log(pT)}+z\right)}{\sqrt{p}} \tag{B.7}$$

with probability at least $1-e^{-z^2}$.

As we have already mentioned in the section devoted to the gradient flow regime, the solution $\Omega(t)$ of (2.14) is exactly the overlap matrix of $W(t)$, and hence the term $\mathcal{R}(W(t))$ can be replaced by $\mathcal{R}(\Omega(t))$ in (B.7).

Rotation invariance The crux of Theorem 2.2 lies in the rotation invariance of μ_0. Indeed, as noticed in [AAM22; HC23], such symmetries are conserved throughout the mean-field dynamics.

Proposition B.1 ([HC23], Proposition 2.1)**.** *Let U be a linear transformation, and assume that the initial measure μ_0, the teacher function $f_\star$ and the data measure ρ are all invariant under U. Let μ_t be the solution to* (GF-PDE) *with initial condition μ_0. Then μ_t is U-invariant for all $t \geq 0$.*

Write $\mathbb{R}^d = V^\star \oplus V^\perp$, where $V^\star$ is the span of $W^\star$ and $V^\perp$ is its orthogonal subspace. Proposition B.1 then implies immediately that μ_t is rotation-invariant on $V^\perp$. Hence, if we define the following function:

$$h(\boldsymbol{w}) = \left(\frac{W^\star\boldsymbol{w}}{d}, \frac{\|\boldsymbol{w}^\perp\|^2}{d}\right) \tag{B.8}$$

where $\boldsymbol{w}^\perp$ is the projection of $\boldsymbol{w}$ on $V^\perp$, then μ_t is only determined by its pushforward $\tilde{\mu}_t = h_\# \mu_t$. Conversely, for a set of reduced parameters $(\boldsymbol{m}, q) \in \mathbb{R}^{k+1}$, and $\boldsymbol{g} \in \mathbb{S}^{d-k-1}$, we define

$$\tilde{\boldsymbol{w}} = W^{\star\top} P^{-1} \boldsymbol{m} + \sqrt{dq} \cdot \boldsymbol{g}, \tag{B.9}$$

where $\boldsymbol{g}$ is a uniform unit vector in $V^\perp$. Then μ_t is the measure of $\tilde{\boldsymbol{w}}$ when $(\boldsymbol{m}, q) \sim \tilde{\mu}_t$ and $\boldsymbol{g} \sim \text{Unif}(\mathbb{S}^{d-k-1})$. This allows us to write the reduced equations for $\tilde{\mu}_t$:

$$\begin{aligned}
\partial_t \tilde{\mu}_t &= \nabla_{(\boldsymbol{m},q)} \cdot \left(\tilde{\mu}_t \tilde{\varphi}(\cdot, \tilde{\mu}_t)\right) \\
\tilde{\varphi}_{\boldsymbol{m}}((\boldsymbol{m}, q), \tilde{\mu}) &= \mathbb{E}_{\boldsymbol{x},\boldsymbol{g}} \left[\sigma'(\tilde{\boldsymbol{w}}^\top \boldsymbol{x}) W^\star \boldsymbol{x} \left(f_{\Theta^\star}(\boldsymbol{x}) - \tilde{f}_{\tilde{\mu}}(\boldsymbol{x})\right)\right] \\
\tilde{\varphi}_q((\boldsymbol{m}, q), \tilde{\mu}) &= \mathbb{E}_{\boldsymbol{x},\boldsymbol{g}} \left[\sigma'(\tilde{\boldsymbol{w}}^\top \boldsymbol{x}) \sqrt{q} \boldsymbol{g}^\top \boldsymbol{x} \left(f_{\Theta^\star}(\boldsymbol{x}) - \tilde{f}_{\tilde{\mu}}(\boldsymbol{x})\right)\right]
\end{aligned} \tag{DF-PDE}$$

where $\boldsymbol{x} = (\boldsymbol{z}, \boldsymbol{r}) \sim \mathcal{N}(0, {}^1\!/_d I_d)$ is a normalised Gaussian vector, $\boldsymbol{g} \sim \text{Unif}(\mathbb{S}^{d-k-1})$, and

$$\tilde{f}_{\tilde{\mu}}(\boldsymbol{x}) = \int \sigma(\tilde{\boldsymbol{w}}^\top \boldsymbol{x}) d\tilde{\mu}(\boldsymbol{m}, q) d\nu(\boldsymbol{g}) \tag{B.10}$$

The associated population risk is now

$$\tilde{\mathcal{R}}(\tilde{\mu}) := \mathbb{E}_{\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, {}^1\!/_d I_d)} \left[\frac{1}{2} \left(f_{\Theta^\star}(\boldsymbol{x}) - \tilde{f}_{\tilde{\mu}}(\boldsymbol{x})\right)^2\right]. \tag{B.11}$$

We have therefore shown the following proposition:

Proposition B.2. *Assume that μ_0 is rotation-invariant on $V^\perp$. Consider these two processes:*

- *the solution μ_t of* (GF-PDE) *with initial value μ_0,*
- *the solution $\tilde{\mu}_t$ of* (DF-PDE) *with initial value $h_\# \mu_0$.*

Then, for all $t \geq 0$, we have

$$\hat{\mathcal{R}}(\mu_t) = \tilde{\mathcal{R}}(\tilde{\mu}_t) \tag{B.12}$$

Back to ODEs Consider a population of p particles $(\boldsymbol{m}, q)$ that evolve according to the following equations:

$$\begin{aligned}
\frac{d\boldsymbol{m_i}}{dt} &= \tilde{\varphi}_{\boldsymbol{m}}(\boldsymbol{m_i}(t), \tilde{\mu}_p(t)) \\
\frac{dq_i}{dt} &= \tilde{\varphi}_q(q_i(t), \tilde{\mu}_p(t))
\end{aligned} \tag{DF-ODE}$$

where $\tilde{\mu}_p(t)$ is the empirical distribution of the population:

$$\tilde{\mu}_p(t) = \frac{1}{p} \sum_{i=1}^{p} \delta_{\boldsymbol{m}_i(t), q_i(t)}.$$

Then, by the same arguments as Theorem B.1, we have:

Proposition B.3. *Assume that $\tilde{\mu}_0$ is a K^2/d-subgaussian distribution, and consider the two processes:*

- *the solution $\tilde{\mu}_t$ of* (DF-PDE) *with initial value $\tilde{\mu}_0$,*
- *the solution $\tilde{\Theta}(t) = (\boldsymbol{m_i}(t), q_i(t))_i$ of* (DF-ODE) *with initial conditions $\boldsymbol{m}_i(0), q_i(0) \sim \tilde{\mu}_0$ i.i.d.*

Then for all $T \geq 0$, we have

$$\sup_{t\in[0,T]} \left|\tilde{\mathcal{R}}(\tilde{\Theta}(t)) - \tilde{\mathcal{R}}(\tilde{\mu}_t)\right| \leq \frac{Ce^{CT}\left(\sqrt{\log(pT)} + z\right)}{\sqrt{p}}, \tag{B.13}$$

with probability at least $1 - e^{-z^2}$, where $\tilde{\mathcal{R}}(\Theta(t)) = \tilde{\mathcal{R}}(\tilde{\mu}_p(t))$.

In conclusion, we have shown the following:

Theorem B.2. *Assume that conditions 2.2 and 2.3 are both satisfied, and define*

$$\mathrm{M}^0 = \frac{\mathrm{W}^0\mathrm{W}^{\star\top}}{d}, \quad \mathrm{Q}^0 = \frac{\mathrm{W}^0\mathrm{W}^{0\top}}{d}, \quad \boldsymbol{q}^0 = \mathrm{diag}\left(Q^0 - M^0P^{-1}M^{0\top}\right). \tag{B.14}$$

Consider the two following processes:

- *the solution $\Omega(t)$ of* (2.14) *with initial value Ω^0,*
- *the solution $\tilde{\Theta}(t)$ of* (DF-ODE) *with initial value $(M^0, \boldsymbol{q}^0)$.*

Then for any $T \geq 0$, we have

$$\sup_{t\in[0,T]} \left|\mathcal{R}(\Omega(t)) - \tilde{\mathcal{R}}(\tilde{\Theta}(t))\right| \leq \frac{Ce^{CT}\left(\sqrt{\log(pT)} + z\right)}{\sqrt{p}}, \tag{B.15}$$

with probability at least $1 - e^{-z^2}$.

Matching the equations To show that Theorem B.2 implies Theorem 2.2, we need to show the following:

- the update equations for (2.24) and (DF-ODE) are the same
- the risk definitions for $\tilde{\mathcal{R}}(\tilde{\Theta})$ matches the one from (2.25).

We will only show it for $\tilde{\varphi}_{\boldsymbol{m}}$; the rest is done similarly. Expanding the definition, we have

$$\tilde{\varphi}_{\boldsymbol{m}}((\boldsymbol{m}_i, q_i), \tilde{\mu}_p)_r = \frac{1}{k}\sum_{s=1}^{k}\langle\sigma'(\tilde{\lambda}_i)\lambda_r^\star\sigma(\lambda_s^\star)\rangle - \frac{1}{p}\sum_{j=1}^{p}\langle\sigma'(\tilde{\lambda}_i)\lambda_r^\star\sigma(\tilde{\lambda}_j)\rangle \tag{B.16}$$

where $\tilde{\lambda}_i = \tilde{\boldsymbol{w}}_i^\top \boldsymbol{x}$ and $\langle\cdot\rangle$ denotes the expectation with respect to $\boldsymbol{x}, \boldsymbol{g}$. On the other hand,

$$\Psi^{(M)}(\Omega) = \frac{1}{k}\sum_{s=1}^{k}\langle\sigma'(\lambda_i)\lambda_r^\star\sigma(\lambda_s^\star)\rangle - \frac{1}{p}\sum_{j=1}^{p}\langle\sigma'(\lambda_i)\lambda_r^\star\sigma(\lambda_j)\rangle, \tag{B.17}$$

without any expectation on $\boldsymbol{g}$; hence we only need to match the expressions term by term. The first sums of (B.16) and (B.17) are actually identical (since the marginal distribution of $\lambda_i^\star$ is independent from $\boldsymbol{g}$), so we look at the second ones:

$$\langle \sigma'(\tilde{\lambda}_i)\lambda_r^\star \sigma(\tilde{\lambda}_j)\rangle = \int \mathbb{E}_{x\sim\mathcal{N}(\mathbf{0},{}^1\!/_d I_d)}\left[\sigma'(\tilde{\lambda}_i)\lambda_r^\star\sigma(\tilde{\lambda}_j)\right] d\nu(\boldsymbol{g}_i)d\nu(\boldsymbol{g}_j) \tag{B.18}$$

For $i \neq j$ and a given realization of $\boldsymbol{g}_i, \boldsymbol{g}_j$, the covariance matrix of $(\tilde{\lambda}_i, \tilde{\lambda}_j, \lambda_r^\star)$ is

$$\mathrm{Cov}\left[\tilde{\lambda}_i, \tilde{\lambda}_j, \lambda_r^\star\right] = \begin{pmatrix} \boldsymbol{m}_i^\top P^{-1}\boldsymbol{m_i} + q_i & \boldsymbol{m}_i^\top P^{-1}\boldsymbol{m_j} + \sqrt{q_i q_j}\xi_{ij} & m_{ir} \\ \boldsymbol{m}_i^\top P^{-1}\boldsymbol{m_j} + \sqrt{q_i q_j}\xi_{ij} & \boldsymbol{m}_i^\top P^{-1}\boldsymbol{m_i} + q_i & m_{jr} \\ m_{ir} & m_{jr} & p_{rr} \end{pmatrix}$$

where $\xi_{ij} = \langle \boldsymbol{g_i}, \boldsymbol{g_j}\rangle$. It is easy to check that this is exactly equal to $\tilde{\Omega}^{ijr}$, where $\tilde{\Omega}$ is defined in (2.23). Finally, by linearity of expectation, since we never have a three-way correlation term between $\boldsymbol{g}_i, \boldsymbol{g}_j, \boldsymbol{g}_\ell$, we can consider all the ξ_{ij} to be independent. This ends the proof of Theorem 2.2.

B.4 Proofs for the high-dimensional mean-field approximation

B.4.1 Preliminaries

We first provide several bounds on expectations of functions of Gaussians, that will be used throughout this section. We begin by recalling the classical Gaussian Poincaré inequality (see e.g. [BLM13], Theorem 3.20):

Lemma B.1. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function, and $X \sim \mathcal{N}(0, I)$. Then*

$$\mathrm{Var}(f(X)) \leq \mathbb{E}\left[\|\nabla f(X)\|^2\right].$$

If $X \sim \mathcal{N}(0, \Sigma)$, the following bounds hold instead:

$$\mathrm{Var}(f(X)) \leq \mathbb{E}\left[\langle \nabla f(X), \Sigma\nabla f(X)\rangle\right] \leq \|\Sigma\|_{\mathrm{op}} \cdot \mathbb{E}\left[\|\nabla f(X)\|^2\right].$$

Now, we provide some concentration bounds for expectations of random variables. Let $h : \mathbb{R} \to \mathbb{R}$ be a function, and z a random variable; our goal is to bound

$$\left|\mathbb{E}[h(z)] - h(\mathbb{E}[z])\right|$$

under different assumptions on f, z.

Lemma B.2. *Assume that h is L-Lipschitz, and that z has a second moment. Then*

$$\left|\mathbb{E}[h(z)] - h(\mathbb{E}[z])\right| \leq L\sqrt{\mathrm{Var}(z)} \tag{B.19}$$

Proof. By Jensen's inequality, we have

$$\begin{aligned}\left|\mathbb{E}[h(z)] - h(\mathbb{E}[z])\right| &\leq \mathbb{E}\left[|h(z) - h(\mathbb{E}[z])|\right] \\ &\leq L\mathbb{E}[|z - \mathbb{E}[z]|] \\ &\leq L\sqrt{\mathrm{Var}(z)}\end{aligned}$$

where the last line uses the Cauchy-Schwarz inequality. □

Lemma B.3. *Assume that h is twice differentiable, with $\left\|h^{(k)}\right\|_\infty \leq K$, and that z has a fourth moment. Then*

$$\left|\mathbb{E}[h(z)] - h(\mathbb{E}[z])\right| \leq \frac{K}{2}\sqrt{\mathbb{E}\left[(z-s)^4\right]} \tag{B.20}$$

Proof. For simplicity, let $s = \mathbb{E}[z]$. By the Lagrange formula for Taylor series, we can write

$$h(x) = h(s) + (x-s)h'(s) + (x-s)^2 R(x) \tag{B.21}$$

where R is bounded by $K/2$. Plugging z into this equation,

$$\left|\mathbb{E}[h(z)] - h(s)\right| = \left|\mathbb{E}\left[(z-s)^2 R(z)\right]\right| \leq \sqrt{\mathbb{E}\left[(z-s)^4\right]\mathbb{E}\left[R(z)^2\right]} \tag{B.22}$$

via the Cauchy-Schwarz inequality, and the result ensues. □

B.4.2 Proof of Lemma 2.1

We only show the result for $\Psi^{(M)}$; the one for $\Psi^\perp$ ensues from similar methods. Recalling the expansion in (B.17) as a sum of 3-point correlation functions, we define the following function of 3×3 matrices:

$$f(\Sigma) = \mathbb{E}_{\boldsymbol{z}\sim\mathcal{N}(\mathbf{0},\Sigma)}\left[\sigma'(z_1)z_2\sigma(z_3)\right] \tag{B.23}$$

Then $\Psi^{(M)}(\Omega)$ is simply an average of $f(\Sigma)$ for Σ submatrices of Ω. We first show the following bound:

Lemma B.4. *Under Assumption 2.2, the function f satisfies the following inequality: for any Σ, Σ' such that $\left\|\Sigma' - \Sigma\right\|_\infty \leq K$,*

$$\left|f(\Sigma') - f(\Sigma)\right| \leq C(1 + \sqrt{\Sigma_{22}})\left\|\Sigma' - \Sigma\right\|_\infty \tag{B.24}$$

Proof. Let Σ and Σ' be two positive semidefinite matrices, and $\delta\Sigma = \Sigma' - \Sigma$. We can write $\delta\Sigma = \delta\Sigma_1 - \delta\Sigma_2$ where both $\delta\Sigma_i$ are positive, and

$$\|\delta\Sigma_i\|_\infty \leq C\|\delta\Sigma_i\|_{\mathrm{op}} \leq C\|\delta\Sigma\|_{\mathrm{op}} \leq C'\|\delta\Sigma\|_\infty$$

using norm equivalence. Hence, we shall assume from now on that $\delta\Sigma$ is positive semidefinite, and write

$$z' = z + \delta z$$

where $\delta z \sim \mathcal{N}(\mathbf{0}, \delta\Sigma)$ is independent from z. Define

$$\delta\sigma = \sigma(z_3') - \sigma(z_3) \quad \text{and} \quad \delta\sigma' = \sigma'(z_1') - \sigma'(z_1)$$

Then

$$\begin{aligned} f(\Sigma') - f(\Sigma) = \quad & \mathbb{E}\left[\sigma'(z_1)z_2\delta\sigma + \sigma'(z_1)\delta z_2\sigma(z_3) + \delta\sigma' z_2\sigma(z_3)\right] \\ & + \mathbb{E}\left[\delta\sigma'\delta z_2\sigma(z_3) + \delta\sigma' z_2\delta\sigma + \sigma'(z_1)\delta z_2\delta\sigma\right] \\ & + \mathbb{E}[\delta\sigma'\delta z_2\delta\sigma] \end{aligned}$$

Since δz is independent from z, and σ satisfies Assumption 2.2, we can apply Lemma B.3 to get

$$|\mathbb{E}_{\delta z_3}[\delta\sigma]| \leq C\delta\Sigma_{33},$$

and hence by the Cauchy-Schwarz inequality

$$\left|\mathbb{E}\left[\sigma'(z_1)z_2\delta\sigma\right]\right| \leq C'\sqrt{\Sigma_{22}}\delta\Sigma_{33}.$$

A similar bound holds for the two other terms of the first line. For the second line, another application of Cauchy-Schwarz yields

$$\begin{aligned} \mathbb{E}_{\delta z}\left[\delta\sigma' z_2\delta\sigma\right] &\leq z_2\sqrt{\mathbb{E}[(\delta\sigma)^2]}\sqrt{\mathbb{E}[(\delta\sigma')^2]} \\ &\leq z_2\sqrt{\delta\Sigma_{11}\delta\Sigma_{33}} \end{aligned}$$

by a combination of Lemmas B.1 and B.2. A similar bound holds for the third line, and it is easily checked that all of the obtained bounds are lower than the one of Lemma B.4. □

Now, the expansion of $\Psi^{(M)}(\tilde{\Omega}) - \Psi^{(M)}(\bar{\Omega})$ is an average of terms of the form $f(\tilde{\Sigma}) - f(\bar{\Sigma})$, where $(\tilde{\Sigma} - \bar{\Sigma})_{ij}$ is either 0 or $\sqrt{q_{ii}^{\perp}q_{jj}^{\perp}}\xi_{ij}$. Hence, we can write each of those terms as $h(\xi_{ij}) - h(0)$, and by Assumption 2.1 and Lemma B.4, the function h is Lipschitz. Lemma 2.1 then ensues from an application of Lemma B.2.

B.4.3 Proof of Equation (2.29)

We begin by showing the following lemma. Recall the definition of $\mathcal{E}$:

$$\mathcal{E} = \frac{1}{k}\sum_{r=1}^{k}\sigma(\lambda_r^{\star}) - \frac{1}{p}\sum_{i=1}^{p}\sigma(\lambda_i) \tag{B.25}$$

Lemma B.5. *There exists a constant $c \geq 0$ such that for any choice of $\boldsymbol{\lambda}^{\star}$,*

$$\mathrm{Var}_{\boldsymbol{\lambda}^{\perp}}(\mathcal{E}) \leq \frac{c\|Q^{\perp}\|_{\mathrm{op}}}{p}$$

Proof. We apply the Gauss-Poincaré inequality of Lemma B.1 to

$$f(\boldsymbol{\lambda}^\perp) = \sum_{i=1}^{p} \sigma([MP^{-1}\boldsymbol{\lambda}^\star]_i + \lambda_i^\perp), \quad \text{which implies} \quad [\nabla f]_i = \sigma'(\lambda_i)$$

Whenever σ is Lipschitz, we thus have $\|\nabla f(\boldsymbol{\lambda}^\perp)\|^2 \leq cp$, and the lemma ensues. □

We are now in a position to show Equation (2.29). For brevity, we denote by $\mathbb{E}$ (resp. $\mathbb{E}_{\text{MF}}$) the expectations with respect to $\boldsymbol{\lambda}^\perp \sim \mathcal{N}(\mathbf{0}, Q^\perp)$ (resp. $\boldsymbol{\lambda}^\perp \sim \mathcal{N}(\mathbf{0}, \text{diag}(Q^\perp))$). Since the marginals of both distributions are the same, and by linearity,

$$\mathbb{E}\left[f(\lambda_i^\perp)\right] = \mathbb{E}_{\text{MF}}\left[f(\lambda_i^\perp)\right] \quad \text{and} \quad \mathbb{E}\left[\mathcal{E}\right] = \mathbb{E}_{\text{MF}}\left[\mathcal{E}\right].$$

Under the distribution $\mathcal{N}(\mathbf{0}, \text{diag}(Q^\perp))$, $\lambda_i^\perp$ is almost independent from $\mathcal{E}$, except for the term containing $\sigma(\lambda_i)$. We can thus write, for any $\boldsymbol{\lambda}^\star$

$$\mathbb{E}_{\text{MF}}\left[\mathcal{E}_i\lambda_i^\perp\right] = \mathbb{E}\left[\sigma'(\lambda_i)\lambda_i^\perp\right]\mathbb{E}\left[\mathcal{E}\right] - \frac{1}{p}\mathbb{E}_{\text{MF}}\left[\sigma'(\lambda_i)\lambda_i^\perp\sigma(\lambda_i)\right] + \frac{1}{p}\mathbb{E}\left[\sigma'(\lambda_i)\lambda_i^\perp\right]\mathbb{E}\left[\sigma(\lambda_i)\right]$$

Hence,

$$\mathbb{E}\left[\mathcal{E}_i\lambda_i^\perp\right] - \mathbb{E}_{\text{MF}}\left[\mathcal{E}_i\lambda_i^\perp\right] = \mathbb{E}\left[\sigma'(\lambda_i)\lambda_i^\perp\left(\mathcal{E} - \mathbb{E}\left[\mathcal{E}\right]\right)\right] - \frac{1}{p}\mathbb{E}\left[\sigma'(\lambda_i)\lambda_i^\perp\left(\sigma(\lambda_i) - \mathbb{E}[\sigma(\lambda_i)]\right)\right] \tag{B.26}$$

Now, using the Cauchy-Schwarz inequality,

$$\left|\mathbb{E}\left[\sigma'(\lambda_i)\lambda_i^\perp\left(\mathcal{E} - \mathbb{E}\left[\mathcal{E}\right]\right)\right]\right| \leq \sqrt{\mathbb{E}\left[\left(\sigma'(\lambda_i)\lambda_i^\perp\right)^2\right]\mathbb{E}\left[\left(\mathcal{E} - \mathbb{E}\left[\mathcal{E}\right]\right)^2\right]}.$$

For Lipschitz σ, the first term is easily bounded by $cQ_{ii}^\perp$, and the second is exactly the variance computed in Lemma B.5. The second term in (B.26) being clearly negligible before the first, we finally get

$$\left|\mathbb{E}\left[\mathcal{E}_i\lambda_i^\perp\right] - \mathbb{E}_{\text{MF}}\left[\mathcal{E}_i\lambda_i^\perp\right]\right| \leq c\sqrt{Q_{ii}^\perp \cdot \frac{\|Q^\perp\|_{\text{op}}}{p}}, \tag{B.27}$$

and Equation (2.29) ensues by taking the expectation w.r.t $\boldsymbol{\lambda}^\star$ on both sides.

B.5 Derivation of explicit expression for the squared activation

In this appendix we show how to derive the differential equations for the dynamics when both σ and σ^* are the square function. We will not present the $\Psi^{(\text{Var})}$ term since we never use the square activation in the high-dimensional regime.

The starting points are Equations (2.12) and the fact that $\mathcal{R} = \mathcal{E}^2/2$. Due to the linearity of the expected value, we can reduce the expectation on products of $\mathcal{E}$ and λ. Let's start with the

population risk. The expected values we need can be expanded to

$$\begin{aligned}
\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\mathcal{E}^2\right] =& \frac{1}{k^2}\sum_{r,s=1}^{k}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma(\lambda_r^*)\sigma(\lambda_s^*)\right] + \\
& \frac{1}{p^2}\sum_{j,l=1}^{p}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma(\lambda_j)\sigma(\lambda_l)\right] \\
& -\frac{2}{pk}\sum_{j=1}^{p}\sum_{r=1}^{k}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma(\lambda_j)\sigma(\lambda_r^*)\right], \\
\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_j)\lambda_l\mathcal{E}\right] =& \frac{1}{k}\sum_{r'=1}^{k}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_j)\lambda_l\sigma(\lambda_{r'}^*)\right] \\
& -\frac{1}{p}\sum_{l'=1}^{p}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_j)\lambda_l\sigma(\lambda_{l'})\right], \\
\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_j)\lambda_r^*\mathcal{E}\right] =& \frac{1}{k}\sum_{r'=1}^{k}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_j)\lambda_r^*\sigma(\lambda_{r'}^*)\right] \\
& -\frac{1}{p}\sum_{l'=1}^{p}\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda_j)\lambda_r^*\sigma(\lambda_{l'})\right].
\end{aligned}$$

These expansions are still valid for any generic activation function. Before specializing in $\sigma(x) = x^2$, we introduce a shorthand in the notation. We will use

$$\omega_{\alpha\beta} \coloneqq []_{\alpha\beta},$$

where the indices α and β can discriminate between teacher and student local fields, as well as the numerical index. Actually, this notation allow us to compute also $\Psi^{\perp}(\Omega)$ by expanding Equation (2.21) as above, and using the matrix in Equation (B.3) as covariance for the normal distribution.

With this consideration, there are only 2 types of expected values to be computed. Let us write them explicitly, using our specific activation function

$$\begin{aligned}
\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma(\lambda^\alpha)\sigma(\lambda^\beta)\right] &= \mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[(\lambda^\alpha)^2(\lambda^\beta)^2\right], \\
\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\sigma'(\lambda^\alpha)\lambda^\beta\sigma(\lambda^\gamma)\right] &= 2\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\lambda^\alpha\lambda^\beta(\lambda^\gamma)^2\right].
\end{aligned}$$

We are left with some expected values of polynomials of Gaussian variables. Since the local fields all have zero mean, these are nothing but moments of a Gaussian distribution with multiple variables. The standard result used to calculate these is the Isserlis' Theorem:

$$\begin{aligned}
\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[(\lambda^\alpha)^2(\lambda^\beta)^2\right] &= \omega_{\alpha\alpha}\omega_{\beta\beta} + 2\omega_{\alpha\beta}^2, \\
2\mathbb{E}_{(\boldsymbol{\lambda},\boldsymbol{\lambda}^\star)\sim\mathcal{N}(\mathbf{0}_{p+k},\Omega)}\left[\lambda^\alpha\lambda^\beta(\lambda^\gamma)^2\right] &= 2\omega_{\alpha\beta}\omega_{\gamma\gamma} + 4\omega_{\alpha\gamma}\omega_{\beta\gamma}.
\end{aligned}$$

By retracing all steps backward and making the necessary substitutions, we can arrive at an explicit form of the Equations (2.14). In order to obtain a matrix form such as in the Equations (2.31), we have to write in a closed form all the summations appeared during the derivation and use the fact that Q and P are symmetric matrices.

B.6 Numerical experiments in the mean-field limit

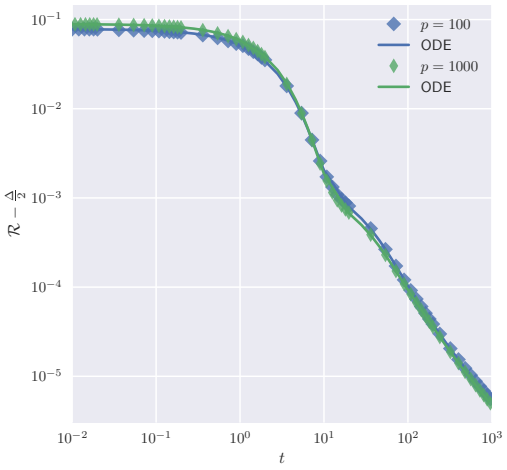
In this appendix we show the result of some numerical experiments we performed to verify the statements exposed in Section 2.3.4. We also refer to our GitHub repository[1].

First, we checked that the point $\mathrm{Q}^{\perp} = 0$ is a fixed point of the dynamic. We compare a simulation of this case with an integration of just the matrix M. Since when $\mathrm{Q}^{\perp} = 0$, M is a *sufficient statistic* by itself, we expect it to be the only parameter to be evolved to characterize the dynamics. Figure B.2 (a) shows agreement between simulation and ODE integration.

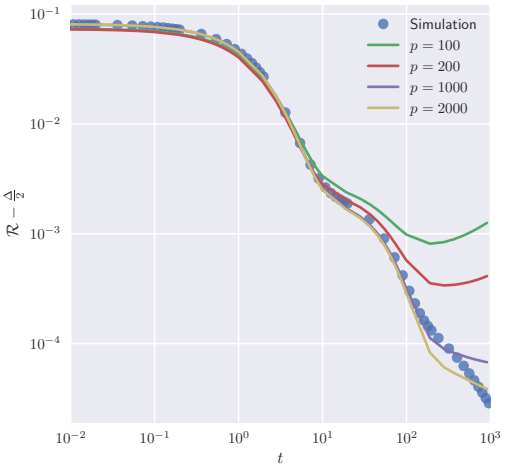
Second, we test whether it is actually necessary to compute the expectation over the matrix Ξ when integrating the mean field in low dimension. In Figure B.2 (b) we present the results for erf activation function: not taking into account the off-diagonal terms, the integration of the ODEs do not match the simulated dynamics even for large p. The effect becomes even more evident by comparing Figure B.2 (c) with Figure 2.3 (a). In fact, we can see that in the case of quadratic activation, neglecting the matrix Ξ leads to a mismatch of population risk even at the initial instant, as it follows naturally from Equation (2.33). We do not see such a marked difference in the case of $\sigma = \mathrm{erf}(\cdot/\sqrt{2})$ since the latter is an odd function and non-diagonal terms of Ξ are symmetric random variables.

Finally, we looked at the evolution of the distribution of student weights. We use $k = 1$ to have just one teacher vector $\boldsymbol{w}^{\star} \in \mathbb{R}^d$, while $p = 5000$ to have enough samples to understand how the distribution is evolving. In particular, we are looking at the distribution of the cosines of the angles between student weights and $\boldsymbol{w}^{\star}$; in Figure B.2 (d) we plot this distribution at 3 different times of the evolution. At $t = 0$, the distribution is uniform, as expected since we sample weights from a spherical invariant distribution with $d = 3$. During the evolution, the student's weight moves in the direction of $\boldsymbol{w}^{\star}$ and $-\boldsymbol{w}^{\star}$ (since σ is an even function the sign of the weights does not count). At large time, the distribution is essentially $1/2\,(\delta_{\boldsymbol{w}^{\star}} + \delta_{-\boldsymbol{w}^{\star}})$, which represents perfect learning.

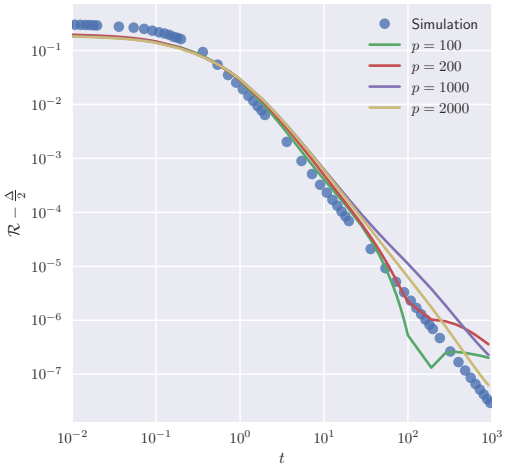
[1] `https://github.com/IdePHICS/DimensionlessDynamicsSGD`



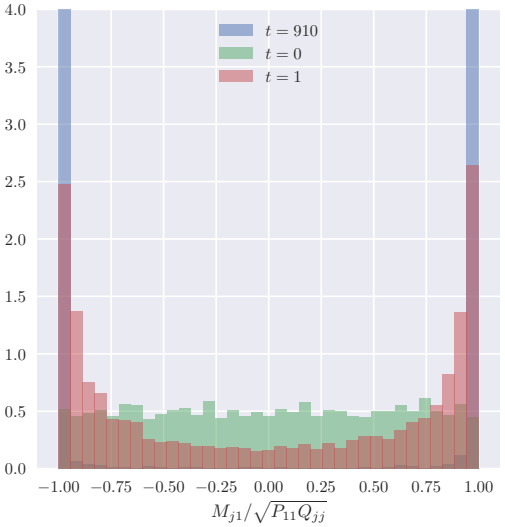
(a) $k = 2, d = 10, \gamma = 1, \Delta = 0, \sigma(x) = \text{erf}(x/\sqrt{2})$



(b) $k = 2, d = 5, \gamma = 1, \Delta = 0, \sigma(x) = \text{erf}(x/\sqrt{2})$



(c) $k = 2, d = 5, \gamma = 1, \Delta = 0, \sigma(x) = x^2$



(d) $k = 1, d = 3, \gamma = 1, \Delta = 0, \sigma(x) = x^2$

Figure B.2: (a) Evolution from starting point $Q^{\perp} = 0$, tracking only M. (b) - (c) Low-dimension simulations and the corresponding ODE trajectory, without the off-diagonal entries of matrix Ξ (d) Histogram of the student-teacher weight overlaps at different times.

Appendix C

Appendices to "Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD"

C.1 Explicit ODEs derivation

C.1.1 Derivation of the process

Let's start by reminding the definition of *displacement* at step ν

$$\mathcal{E}^\nu \coloneqq (\lambda_\star^\nu)^2 - \frac{1}{p}\sum_{j=1}^{p} a_j(\lambda_j^\nu)^2 + \sqrt{\Delta}z^\nu, \tag{C.1}$$

from which it's easy to write the loss function

$$\ell(y^\nu, f_{\Theta^\nu}(x^\nu)) = \frac{1}{2}\left(\mathcal{E}^\nu\right)^2. \tag{C.2}$$

The gradient respect to the parameters is given

$$\begin{aligned}\partial_{a_j}\ell(y^\nu, f_{\Theta^\nu}(x^\nu)) &= -\frac{1}{p}\mathcal{E}^\nu(\lambda_j^\nu)^2\\ \nabla_{\boldsymbol{w}_j}\ell(y^\nu, f_{\Theta^\nu}(x^\nu)) &= -\frac{1}{p}\mathcal{E}^\nu 2a_j^\nu\lambda_j^\nu x^\nu\end{aligned} \tag{C.3}$$

Using γ as learning rate for the weights $\boldsymbol{w}_j$ and γ/d for the second layer, we have the following update equations

$$\begin{aligned}a_j^{\nu+1} &= a_j^\nu + \frac{\gamma}{pd}\mathcal{E}^\nu(\lambda_j^\nu)^2\\ \boldsymbol{w}_j^{\nu+1} &= \boldsymbol{w}_j^\nu + \frac{\gamma}{p}\mathcal{E}^\nu 2a_j^\nu\lambda_j^\nu x^\nu\end{aligned} \tag{C.4}$$

Applying the definition of the sufficient statistics, $m_j = \boldsymbol{w}_j\boldsymbol{w}_\star/d$ and $\mathrm{Q}_{jl} = \boldsymbol{w}_j\boldsymbol{w}_l/d$, we can recover Equations (3.7).

C.1.2 Explicit ODE

To get the explicit form of our ODEs we need to compute some expected value over the preactivations $(\lambda_\star, \lambda)$. These are gaussian variables, whose correlation matrix is given by Ω. Similarly, the population risk is defined as

$$\mathcal{R} = \mathbb{E}_{(\lambda,\lambda^\star)\sim\mathcal{N}(0_{p+1},\Omega)}\left[\frac{1}{2}\mathcal{E}^2\right].$$

Let's look close to the random variable we need for this expected values, expressing them just as function of local fields. To be more concise, from now on with $\mathbb{E}$ we always mean the expected value over $(\lambda, \lambda^\star) \sim \mathcal{N}(0_{p+1}, \Omega)$. For the risk we just need

$$\mathbb{E}\left[\mathcal{E}^2\right] = \lambda_\star^4 + \frac{1}{p^2}\sum_{j,l=1}^{p} a_j a_l \mathbb{E}\left[\lambda_j^2\lambda_l^2\right] - \frac{2}{p}\sum_{j=1}^{p} a_j \mathbb{E}\left[\lambda_j^2\lambda_\star^2\right],$$

while for the ODE of $\bar{a}$

$$\mathbb{E}\left[\mathcal{E}\lambda_j^2\right] = \mathbb{E}\left[\lambda_\star^2\lambda_j^2\right] - \frac{1}{p}\sum_{l=1}^{p} a_l \mathbb{E}\left[\lambda_l^2\lambda_j^2\right],$$

where we omitted the noise part since it averages out with the expectation. The equation for $\bar{m}$ requires

$$2a_j\mathbb{E}\left[\mathcal{E}\lambda_j\lambda_\star\right] = 2\mathbb{E}\left[\lambda_\star^2\lambda_j\lambda_\star\right] - \frac{2}{p}\sum_{l=1}^{p} a_j a_l \mathbb{E}\left[\lambda_l^2\lambda_j\lambda_\star\right],$$

while we need two different expectations for $\bar{\mathrm{Q}}$

$$2(a_j + a_l)\mathbb{E}\left[\mathcal{E}\lambda_j\lambda_\star\right] = 2(a_j + a_l)\left[\mathbb{E}\left[\lambda_\star^2\lambda_j\lambda_l\right] - \frac{1}{p}\sum_{s=1}^{p} a_s \mathbb{E}\left[\lambda_s^2\lambda_j\lambda_l\right]\right] \quad \text{and}$$

$$4\frac{\gamma}{p}\mathcal{E}^2 a_j a_l \lambda_j\lambda_l = 16\frac{\gamma}{p}a_j a_l\left[\mathbb{E}\left[\lambda_j\lambda_l\lambda_\star^4\right] - \frac{2}{p}\sum_{s=1}^{p} a_s\mathbb{E}\left[\lambda_j\lambda_l\lambda_\star^2\lambda_s^2\right] + \right.$$
$$\left. + \frac{1}{p^2}\sum_{s,t=1}^{p}\left(a_s a_t \mathbb{E}\left[\lambda_j\lambda_l\lambda_s^2\lambda_t^2\right] + \Delta\mathbb{E}\left[\lambda_j\lambda_l\right]\right)\right].$$

We are left to compute some distribution moments of a multivariate Gaussian of second, fourth and sixh order. In the GitHub repository[1] can be found a Mathematica script to address this task; alternatively Isserlis' Theorem can be applied. We introduce a shorthand in the notation

$$\omega_{\alpha\beta} \coloneqq [\Omega]_{\alpha\beta},$$

[1] `https://github.com/IdePHICS/EscapingMediocrity`

where the indices α and β can discriminate between local fields λ (if $\alpha, \beta \in [1, \dots, p]$, and $\lambda_\star$ (if $\alpha, \beta = p+1$). The final result are given by

$$\begin{aligned}
\mathbb{E}\left[\lambda_\alpha \lambda_\beta\right] &= \omega_{\alpha\beta} \\
\mathbb{E}\left[\lambda_\alpha^2 \lambda_\beta^2\right] &= \omega_{\alpha\alpha}\omega_{\beta\beta} + 2\omega_{\alpha\beta}^2 \\
\mathbb{E}\left[\lambda_\alpha \lambda_\beta \lambda_\gamma^2\right] &= \omega_{\alpha\beta}\omega_{\gamma\gamma} + 2\omega_{\alpha\gamma}\omega_{\beta\gamma} \\
\mathbb{E}\left[\lambda_\alpha \lambda_\beta \lambda_\gamma^2 \lambda_\delta^2\right] &= \omega_{\alpha\beta}\omega_{\gamma\gamma}\omega_{\delta\delta} + 2\omega_{\alpha\beta}\omega_{\gamma\delta}^2 + 2\omega_{\alpha\gamma}\omega_{\beta\gamma}\omega_{\delta\delta} + \\
&\qquad 4\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\gamma\delta} + 4\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\gamma\delta} + 2\omega_{\alpha\delta}\omega_{\beta\delta}\omega_{\gamma\gamma}
\end{aligned}$$

By retracing all steps backward and making the necessary substitutions, we can arrive at an explicit form of the ODEs and population risk. While the full risk expression can be found in Equation (3.6), we report here just the case $a_j = 1$ for the ODEs since they have a compact matrix form

$$\frac{\mathrm{d}\mathrm{M}}{\mathrm{d}t} = 2\left(\rho - \frac{\operatorname{Tr}[\mathrm{Q}]}{p}\right)\mathrm{M} + 4\left(\rho\mathrm{M} - \frac{\mathrm{Q}\mathrm{M}}{p}\right) \tag{C.5a}$$

$$\begin{aligned}
\frac{\mathrm{d}\mathrm{Q}}{\mathrm{d}t} &= 4\left(\rho - \frac{\operatorname{Tr}[\mathrm{Q}]}{p}\right)\mathrm{Q} + 8\left(\frac{\mathrm{M}\mathrm{M}^\top}{p} - \frac{\mathrm{Q}^2}{p}\right) \\
&\quad + \frac{4\gamma}{p}\Bigg\{\left[3\rho^2\mathrm{Q} + 12\rho\mathrm{M}\mathrm{M}^\top\right] + \frac{1}{p^2}\Bigg[\left(\operatorname{Tr}[\mathrm{Q}]^2 + 2\operatorname{Tr}\left[\mathrm{Q}^2\right]\right)\mathrm{Q} + 4\operatorname{Tr}[\mathrm{Q}]\,\mathrm{Q}^2 + 8\mathrm{Q}^3\Bigg] \\
&\qquad - \frac{2}{p}\Bigg[\left(\rho\operatorname{Tr}[\mathrm{Q}] + 2\operatorname{Tr}\left[\mathrm{M}\mathrm{M}^\top\right]\right)\mathrm{Q} + 2\operatorname{Tr}[\mathrm{Q}]\,\mathrm{M}\mathrm{M}^\top \\
&\qquad\quad + 2\rho\mathrm{Q}^2 + 4\left(\mathrm{M}\mathrm{M}^\top\mathrm{Q} + \mathrm{Q}\mathrm{M}\mathrm{M}^\top\right)\Bigg] + \Delta\mathrm{Q}\Bigg\},
\end{aligned} \tag{C.5b}$$

where $\rho \coloneqq \boldsymbol{w}_\star^2/d$. For completeness, this is Equation (3.6) for the case $a_j = 1$

$$\mathcal{R}(\Omega) = \frac{3+\Delta}{2} - \frac{\rho\operatorname{Tr}[\mathrm{Q}] + 2\operatorname{Tr}\left[\mathrm{M}\mathrm{M}^\top\right]}{p} + \frac{\operatorname{Tr}[\mathrm{Q}]^2 + 2\operatorname{Tr}\left[\mathrm{Q}^2\right]}{2p^2}. \tag{C.6}$$

C.2 Spherically constrainted ODE and SDE

C.2.1 Spherical constraint for ODE

Let's recall the update rule for the weights

$$\boldsymbol{w}_j^{\nu+1} = \frac{\boldsymbol{w}_j^\nu - \gamma\nabla_{\boldsymbol{w}_j}\ell(y^\nu, f_{\Theta^\nu}(x^\nu))}{\left\|\boldsymbol{w}_j^\nu - \gamma\nabla_{\boldsymbol{w}_j}\ell(y^\nu, f_{\Theta^\nu}(x^\nu))\right\|}\sqrt{d}, \tag{C.7}$$

we will find the leading order approximation of it and then apply the argument as the unconstrained case for deriving the ODEs.

To shorten the notation we will use ℓ^ν for indicating $\ell(y^\nu, f_{\Theta^\nu}(x^\nu))$. Let's start by computing the normalization factor

$$\begin{aligned}\frac{1}{\left\|\boldsymbol{w}_j^\nu - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|} &= \left[\left(\boldsymbol{w}_j^\nu - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right)\cdot\left(\boldsymbol{w}_j^\nu - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right)\right]^{-\frac{1}{2}}\\ &= \left[\left\|\boldsymbol{w}_j^\nu\right\|^2 - 2\gamma\boldsymbol{w}_j^\nu\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu + \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2\right]^{-\frac{1}{2}}\\ &= \frac{1}{\sqrt{d}}\left[1 - \frac{1}{d}\left(2\gamma\boldsymbol{w}_j^\nu\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu - \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2\right)\right]^{-\frac{1}{2}}\\ &= \frac{1}{\sqrt{d}}\left[1 + \frac{1}{2d}\left(2\gamma\boldsymbol{w}_j^\nu\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu - \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2\right) + o\left(d^{-1}\right)\right].\end{aligned}$$

Note that we kept both two terms in the expansion because we can show that both the norm and the scalar product with a weight vector are order 1

$$\begin{aligned}\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2 &\sim \frac{2\mathcal{E}^2\lambda_j^\nu}{p^2}\frac{\sum_{i=1}^d\left(x_i^\nu\right)^2}{d} \sim \frac{2\mathcal{E}^2\lambda_j^\nu}{p^2}\frac{\chi_d^2}{d} = \mathcal{O}(1),\\ \boldsymbol{w}_j\cdot\nabla_{\boldsymbol{w}_l}\ell^\nu &\sim \frac{2\mathcal{E}\lambda_l^\nu}{p}\sum_{i=1}^d\frac{w_{j,i}}{\sqrt{d}}\mathcal{N}(0,1) \sim \frac{2\mathcal{E}\lambda_l^\nu}{p^2}\mathcal{N}\left(0,\sum_{i=1}^d\frac{w_{j,i}^2}{d}\right) \sim \frac{2\mathcal{E}\lambda_l^\nu}{p^2}\mathcal{N}(0,1) = \mathcal{O}(1).\end{aligned}$$

We can now plug the expansion back into the original update rule

$$\boldsymbol{w}_j^{\nu+1} = \left(\boldsymbol{w}_j^\nu - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right)\left[1 + \frac{1}{2d}\left(2\gamma\boldsymbol{w}_j^\nu\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu - \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2\right) + o\left(d^{-1}\right)\right]\sqrt{d}.$$

We are ready to go over the steps that take us from the update rule on vector weights to those on order parameters. We report by way of example the steps performed for M; the accounts for Q are similar, just a bit more tedious.

$$\begin{aligned}\mathrm{M}_j^{\nu+1} &= \frac{\boldsymbol{w}_j^{\nu+1}\cdot\boldsymbol{w}^*}{d}\\ &= \left(\mathrm{M}_j^\nu - \frac{\gamma\boldsymbol{w}^*\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu}{d}\right)\left[1 + \frac{1}{2d}\left(2\gamma\boldsymbol{w}_j^\nu\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu - \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2\right) + o\left(d^{-1}\right)\right]\\ &= \mathrm{M}_j^\nu - \frac{\gamma\boldsymbol{w}^*\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu}{d} + \frac{\mathrm{M}_j^\nu}{2d}\left(2\gamma\boldsymbol{w}_j^\nu\cdot\nabla_{\boldsymbol{w}_j}\ell^\nu - \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2\right) + o\left(d^{-1}\right)\\ &= \mathrm{M}_j^\nu + \frac{1}{d}\left[\frac{\gamma}{p}\lambda_\star\mathcal{E}^\nu\lambda_j^\nu - \frac{\mathrm{M}_j^\nu}{2}\left(2\frac{\gamma}{p}\mathcal{E}^\nu\lambda_j^\nu\lambda_j^\nu + \frac{\gamma^2}{p^2}\mathcal{E}^\nu\lambda_j^{\nu 2}\right)\right] + o\left(d^{-1}\right).\end{aligned} \tag{C.8}$$

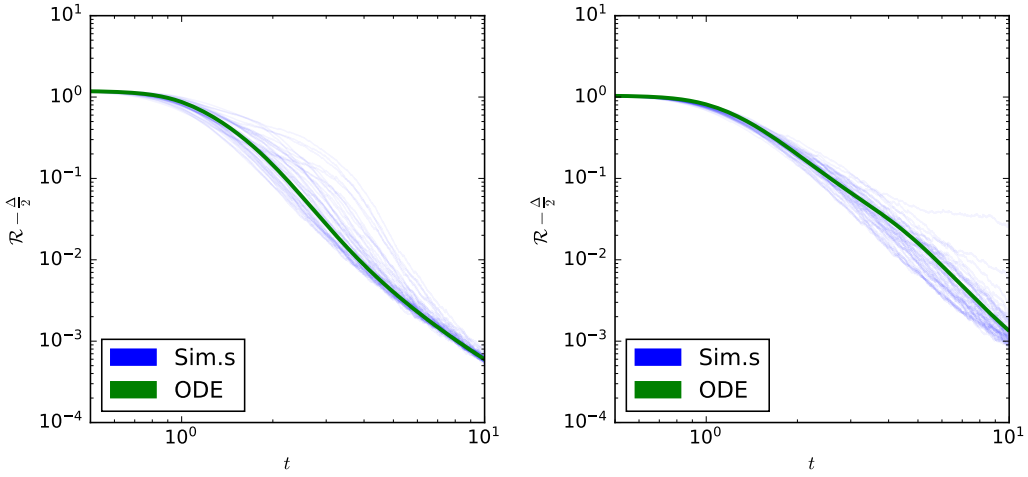


Figure C.1: comparison of ODE integration and many SGD runs for $p = 5$ (left) and $p = 20$ (right). Both the experiments have $d = 5000$.

We can now take the limit $d \to +\infty$, claiming that the theorem [Vei+22; Gol+19] proving ODE convergence is still valid. Indeed, the error term $o(d^{-1})$ in Equation (C.8) has an average order of $O(d^{-2})$, which can be absorbed in the term Γ^{ν} of Theorem A.1 in [Vei+22]. The rest of the proof proceeds the same way, noting that the square function can be assumed to be Lipschitz since the dynamics take place on the sphere.

The differential equation that describes the evolution of M is

$$\frac{\mathrm{d}\bar{\mathrm{M}}_j(t)}{\mathrm{d}t} = \mathbb{E}_{\boldsymbol{\lambda},\boldsymbol{\lambda}^* \sim \boldsymbol{\mathcal{N}}(\mathbf{0},\boldsymbol{\Omega}(t))} \left[2\mathcal{E}\lambda_j\lambda_\star - \frac{\bar{\mathrm{M}}_j(t)}{2} \left(4\mathcal{E}\lambda_j\lambda_j + 4\frac{\gamma}{p}\mathcal{E}^2\lambda_j^2 \right) \right];$$

Using the definitions introduced in Equations (3.15) we can write the equation in a nicer form

$$\frac{\mathrm{d}\bar{\mathrm{M}}_j(t)}{\mathrm{d}t} = \Psi_j(\boldsymbol{\Omega}) - \frac{\bar{\mathrm{M}}_j(t)}{2}\Phi_{jj}(\boldsymbol{\Omega}). \tag{C.9}$$

Essentially, the spherical constraint can be imposed by using a term proportional to the unconstrained Q update.

Without reporting all the calculations, we can write an analogous differential equation for Q evolution

$$\frac{\mathrm{d}\bar{\mathrm{Q}}_{jl}(t)}{\mathrm{d}t} = \Phi_{jl}(\boldsymbol{\Omega}) - \frac{\bar{\mathrm{Q}}_{jl}(t)}{2}\left(\Phi_{jj}(\boldsymbol{\Omega}) + \Phi_{ll}(\boldsymbol{\Omega})\right). \tag{C.10}$$

Note that $\dfrac{\mathrm{d}\bar{\mathrm{Q}}_{jj}(t)}{\mathrm{d}t} = 0$ if $\mathrm{Q}_{jj}(t) = 1$, as it should be since the norm of spherical vectors must not change.

In Figure C.1 we show two examples of integration of ODEs, for different values of p. Simulating for large but finite d does not kills the stochasticity in the SGD runs, but we can clearly see how the ODE well describe the dynamics on average.

C.2.2 Spherical constraint for SDE

This subsection we derive the spherical constraint for the SDE, with $p = 1$. We assume that the stochastic process is given

$$\begin{aligned} \mathrm{dM} &= \Psi_1(\Omega)\,\mathrm{d}t + \sqrt{\frac{\gamma}{d}}\boldsymbol{\sigma}_{\mathrm{M}}(\Omega)\cdot \mathrm{d}B_t \\ \mathrm{dQ} &= \Phi_{11}(\Omega)\,\mathrm{d}t + \sqrt{\frac{\gamma}{d}}\boldsymbol{\sigma}_{\mathrm{Q}}(\Omega)\cdot \mathrm{d}B_t, \end{aligned} \tag{C.11}$$

without providing explicit expressions for $\boldsymbol{\sigma}_{\mathrm{M}}(\Omega)$ and $\boldsymbol{\sigma}_{\mathrm{Q}}(\Omega)$; see Appendix C.5 for that.

The derivation is basically following the steps of the previous Section. Starting from the unconstrainted update rule for the weights

$$\boldsymbol{w}_j^{\nu+1} = \boldsymbol{w}_j^{\nu} - \gamma \nabla_{\boldsymbol{w}_j}\ell^{\nu},$$

we can find an expression for the two unconstrained differentials

$$\begin{aligned} \mathrm{d}q &= \frac{-2\gamma \boldsymbol{w}\cdot\nabla\ell^{\nu} + \gamma^2\left\|\nabla\ell^{\nu}\right\|^2}{d} \\ \mathrm{d}m &= \frac{-\gamma \boldsymbol{w}_\star\cdot\nabla\ell^{\nu}}{d}. \end{aligned} \tag{C.12}$$

Since we are forcing the weight on the sphere, the update rule that actually has to be used is

$$\boldsymbol{w}_j^{\nu+1} = \frac{\boldsymbol{w}_j^{\nu} - \gamma\nabla_{\boldsymbol{w}_j}\ell^{\nu}}{\left\|\boldsymbol{w}_j^{\nu} - \gamma\nabla_{\boldsymbol{w}_j}\ell^{\nu}\right\|}\sqrt{d};$$

multiplying both sides by $\boldsymbol{w}_\star$ and subtracting m we get

$$\mathrm{d}m_S = \frac{m + \mathrm{d}m}{\|\boldsymbol{w} - \gamma\nabla\ell^{\nu}\|}\sqrt{d} - m,$$

where we introduce m_S to differentiate the constrained variable from m. Let's estimate the normalization factor

$$\begin{aligned} \|\boldsymbol{w} - \gamma\nabla\ell^{\nu}\| =& \sqrt{(\boldsymbol{w} - \gamma\nabla\ell^{\nu})^2} = \sqrt{\boldsymbol{w}^2 - 2\boldsymbol{w}\cdot\gamma\nabla\ell^{\nu} + \gamma^2\left\|\nabla\ell^{\nu}\right\|^2} \\ =& \sqrt{d}\sqrt{\frac{\boldsymbol{w}^2}{d} + \frac{-2\boldsymbol{w}\cdot\gamma\nabla\ell^{\nu} + \gamma^2\left\|\nabla\ell^{\nu}\right\|^2}{d}} = \sqrt{d}\sqrt{q + \mathrm{d}q} \\ =& \sqrt{d}\sqrt{1 + \mathrm{d}q}, \end{aligned}$$

where in the last step we used the constraint $q = 1$. We can now plug it back in $\mathrm{d}m_S$

$$\mathrm{d}m_S = \frac{m + \mathrm{d}m}{\sqrt{1 + \mathrm{d}q}} - m,$$

and expanding up to leading orders we get

$$\begin{aligned}\mathrm{d}m_S =&(m+\mathrm{d}m)(1+\mathrm{d}q)^{-\frac{1}{2}} - m = (m+\mathrm{d}m)\left(1-\frac{1}{2}\,\mathrm{d}q+\frac{3}{8}\,\mathrm{d}q^2\right) - m\\ &= \mathrm{d}m - \frac{m}{2}\,\mathrm{d}q - \frac{1}{2}\,\mathrm{d}m\,\mathrm{d}q + \frac{3}{8}m\,\mathrm{d}q^2\end{aligned}$$

In principle, we can now use the Itô Lemma on differentials Equations (C.11), obtaining

$$\mathrm{d}m^2 = \frac{\gamma}{d}\boldsymbol{\sigma}_{\mathrm{M}}^2(\Omega)\,\mathrm{d}t, \quad \mathrm{d}q^2 = \frac{\gamma}{d}\boldsymbol{\sigma}_{\mathrm{Q}}^2(\Omega)\,\mathrm{d}t, \quad \text{and} \quad \mathrm{d}m\,\mathrm{d}q = \frac{\gamma}{d}\boldsymbol{\sigma}_{\mathrm{M}}(\Omega)\cdot\boldsymbol{\sigma}_{\mathrm{Q}}(\Omega)\,\mathrm{d}t.$$

It's interesting to note that these lead to a drift correction (and not just stochastic), but it's of second order. As expected, in numerical simulations we can't see the effect for these corrections, hence we neglet them in what follows. Finally, we write the explicit Brownian motion for the constrained dynamic

$$\mathrm{d}m_S = \left(\Psi_1(\Omega) - \frac{m_S}{2}\Phi_{11}(\Omega)\right)\mathrm{d}t + \sqrt{\frac{\gamma}{d}}\left(\boldsymbol{\sigma}_{\mathrm{M}}(\Omega) - \frac{m_S}{2}\boldsymbol{\sigma}_{\mathrm{Q}}(\Omega)\right)\cdot\mathrm{d}B_t. \tag{C.13}$$

Of course, all functions depending on Ω should be evaluated at $m = m_S, q = 1$.

C.3 Derivation of the expected exit time formulas

C.3.1 Linearization of the equations

The linear approximation of Ψ around $m_j \approx 0$ is given by

$$\Psi_j = 4\left(m_j - \frac{m_j}{p}\right) = 4\left(1-\frac{1}{p}\right)m_j, \tag{C.14}$$

while for Φ we distinguish the cases $j = l$ or not

$$\begin{aligned}j \neq l \quad \Phi_{jl} &= 4\left[2\left(-2\frac{\mathrm{Q}_{jl}}{p}\right)\right] +\\ &\quad + \frac{4\gamma}{p}\left\{3\mathrm{Q}_{jl} - \frac{2}{p}\left[p\mathrm{Q}_{jl} + 4\mathrm{Q}_{jl}\right] +\right.\\ &\quad \left.\frac{1}{p^2}\left[\left(p^2+2p\right)\mathrm{Q}_{jl} + 8p\mathrm{Q}_{jl} + 24\mathrm{Q}_{jl}\right] + \Delta\mathrm{Q}_{jl}\right\}\\ &= -\frac{16}{p}\mathrm{Q}_{jl}\\ &\quad + \frac{4\gamma}{p}\left\{3 - 2 - \frac{8}{p} + 1 + \frac{2}{p} + \frac{8}{p} + \frac{24}{p^2} + \Delta\right\}\mathrm{Q}_{jl}\\ &= -\frac{16}{p}\mathrm{Q}_{jl}\\ &\quad + \frac{4\gamma}{p}\left\{2 + \frac{2}{p} + \frac{24}{p^2} + \Delta\right\}\mathrm{Q}_{jl}\end{aligned} \tag{C.15}$$

$$
\begin{aligned}
j = l \quad \Phi_{jj} &= 4\left[2\left(-\frac{1}{p}\right)\right] + \frac{4\gamma}{p}\left\{3 - \frac{2}{p}\left[p+2\right] + \frac{1}{p^2}\left[\left(p^2+2p\right) + 4p + 8\right] + \Delta\right\} \\
&= -\frac{8}{p} + \frac{4\gamma}{p}\left\{3 - 2 - \frac{4}{p} + 1 + \frac{2}{p} + \frac{4}{p} + \frac{8}{p^2} + \Delta\right\} \\
&= -\frac{8}{p} + \frac{4\gamma}{p}\left\{2 + \frac{2}{p} + \frac{8}{p^2} + \Delta\right\}
\end{aligned}
\tag{C.16}
$$

Given these linear approximations, we are ready to write down the equations valid as long as the risk stays in the first plateau

$$
\begin{aligned}
\frac{\mathrm{d}\left[\mathrm{M}(t)\right]_j}{\mathrm{d}t} &= \left[4\left(1 - \frac{1}{p}\right) + \frac{4}{p} - \frac{2\gamma}{p}\left(2 + \frac{2}{p} + \frac{8}{p^2} + \Delta\right)\right]\left[\mathrm{M}(t)\right]_j \\
&= \left[4 - \frac{2\gamma}{p}\left(2 + \frac{2}{p} + \frac{8}{p^2} + \Delta\right)\right]\left[\mathrm{M}(t)\right]_j \\
&= 4\left[1 - \frac{\gamma}{p}\left(1 + \frac{1}{p} + \frac{4}{p^2} + \frac{\Delta}{2}\right)\right]\left[\mathrm{M}(t)\right]_j,
\end{aligned}
\tag{C.17}
$$

$$
\begin{aligned}
\frac{\mathrm{d}\left[\mathrm{Q}(t)\right]_{jl}}{\mathrm{d}t} &= \left[-\frac{16}{p} + \frac{4\gamma}{p}\left(2 + \frac{2}{p} + \frac{24}{p^2} + \Delta\right) + \frac{8}{p} - \frac{4\gamma}{p}\left(2 + \frac{2}{p} + \frac{8}{p^2} + \Delta\right)\right]\left[\mathrm{Q}(t)\right]_{jl} \\
&= \left[-\frac{8}{p} + \frac{4\gamma}{p}\left(\frac{16}{p^2}\right)\right]\left[\mathrm{Q}(t)\right]_{jl} \\
&= -\frac{8}{p}\left[1 - \frac{8\gamma}{p^2}\right]\left[\mathrm{Q}(t)\right]_{jl}.
\end{aligned}
\tag{C.18}
$$

We observe that the evolution of the sufficient statistics is uncoupled in the starting saddle. We can shorthand the notation by introducing ω_Q and ω_M

$$
\begin{aligned}
\frac{\mathrm{dM}_j}{\mathrm{d}t} &= \omega_M \mathrm{M}_j \\
\frac{\mathrm{dQ}_{jl}}{\mathrm{d}t} &= -\omega_Q \mathrm{Q}_{jl} \quad \text{when } j \neq l.
\end{aligned}
\tag{C.19}
$$

These equations admit a simple solution given by

$$
\begin{aligned}
\mathrm{M}_j(t) &= \mathrm{M}_j(0)\exp[\omega_M t] \\
\mathrm{Q}_{jl}(t) &= \mathrm{Q}_{jl}(0)\exp[-\omega_Q t].
\end{aligned}
\tag{C.20}
$$

C.3.2 Solving the approximated risk equation

From Equation (C.6) when the weights are on the sphere, it follows that the risk is given by

$$
\mathcal{R}(\mathrm{Q},\mathrm{M}) - \frac{\Delta}{2} = 1 + \frac{1}{p} + \frac{1}{p^2}\sum_{j,l=1;j\neq l}^{p} \mathrm{Q}_{jl}^2 - \frac{2}{p}\sum_{j=1}^{p}\mathrm{M}_j^2 \tag{C.21}
$$

Equations (C.20) can be used to obtain a deterministic time evolution of the risk. The only source of randomness left is from the initial conditions, we can define two random variables as

$$\frac{\mu_0}{d} \coloneqq \sum_{j=1}^{p} \left[\mathrm{M}_j(0)\right]^2 \quad \text{and} \quad \frac{\tau_0}{d} \coloneqq \sum_{j,l=1;j\neq l}^{p} \left[\mathrm{Q}_{jl}(0)\right]^2, \tag{C.22}$$

and get an expression for the risk in function of time

$$\mathcal{R}(t) - \frac{\Delta}{2} = 1 + \frac{1}{p} + \frac{\tau_0}{dp^2}\exp[-2\omega_Q t] - \frac{2\mu_0}{dp}\exp[2\omega_M t] \tag{C.23}$$

This equation is not polished enough to be solved analytically yet. First of all we need to assume that the risk is decreasing by forcing $\omega_{\mathrm{M}} > 0$. Secondly, we want the exponential proportional to τ_0 to be negligible when $t > 0$: this follow from $\omega_{\mathrm{Q}} > 0$. In principle, this last condition is not needed for the process to converge like the first one, but without it is not possible to find an analytical solution for the cross -threshold equation. Moreover, when $p > 6$: $\omega_{\mathrm{M}} > 0 \implies \omega_{\mathrm{Q}} > 0$, so we can see that the request is not unreasonable.

Wrapping all this consideration together, Equation (3.21) is

$$(1-T)\left(1 + \frac{1}{p} + \frac{\tau_0}{dp^2} - \frac{2\mu_0}{dp}\right) = 1 + \frac{1}{p} - \frac{2\mu_0}{dp}\exp[2\omega_M t_{\text{ext}}], \tag{C.24}$$

from where we can compute the exit time

$$t_{\text{ext}} = \frac{\log\left[\frac{Tp(p+1)d + (2\mu_0 p - \tau_0)(1-T)}{2\mu_0 p}\right]}{2\omega_{\mathrm{M}}}.$$

C.3.3 Averaging on initial conditions

As of now t_{ext} is still depending on the initial conditions through the random variables μ_0 and τ_0. Following from the choosen initial conditions, and taking into account the dependence between the two random variables

$$\mu_0, \tau_0 \sim \mathcal{P}_p^d \quad \text{where } \mathcal{P}_p^d \equiv \left(d\sum_{j=1}^{p}(u_j \cdot v)^2, 2d\sum_{j=1}^{p}\sum_{l=j+1}^{p}(u_j \cdot u_l)^2\right) \text{ with } v, u_j \sim \mathbb{S}^{d-1}(1).$$

We have now two possibility to get the expectation of t_{ext}. The first one is known in statistical physics literature as *quenched formula* leaves us with an unexpressed expected value

$$t_{\text{ext}}^{(\text{qnc})} = \mathbb{E}_{\mu_0,\tau_0\sim\mathcal{P}_p^d} \left[\frac{\log\left[\frac{Tp(p+1)d + (2\mu_0 p - \tau_0)(1-T)}{2\mu_0 p}\right]}{2\omega_{\mathrm{M}}}\right]. \tag{C.25}$$

The second one, often referred as the *annealed formula*, is obtain by simply replacing the random variables with their expected values

$$t_{\text{ext}}^{(\text{anl})} = \frac{\log\left[\frac{T(p+1)d + (p+1)(1-T)}{2p}\right]}{2\omega_{\mathrm{M}}}. \tag{C.26}$$

Case $p = 1$ For completeness, let's reduce these formulas to the simplest case $p = 1$. In this case τ_0 does not appear, and from Equation (3.13) we find that $\mu_0 \sim \chi^2(1)$, so the quenched formula reduces to

$$t_{\text{ext}}^{(\text{qnc})} = \mathbb{E}_{\mu_0 \sim \chi^2(1)} \left[\frac{\log \left[\frac{Td}{\mu_0} + (1-T) \right]}{8(1-6\gamma) - 4\gamma\Delta} \right]. \tag{C.27}$$

C.3.4 Overparameterization Gain

Let us introduce the number of gradient steps needed to escape the threshold. Remembering $t = \nu\gamma/pd$, we define

$$s_{\text{ext}}(p, d, \gamma, \Delta, T) \coloneqq \frac{pd}{\gamma} t_{\text{ext}}. \tag{C.28}$$

In the domain of our interest, namely $s_{\text{ext}} > 0$, s_{ext} is a convex function in γ. Therefore, there exists a unique minimum that corresponds to the minimum number of steps required to cross the threshold when p, d are fixed

$$0 = \left. \frac{\partial s_{\text{ext}}(p, d, \gamma, \Delta, T)}{\partial \gamma} \right|_{\gamma = \gamma_{\text{opt}}(p,d,\Delta)} \qquad \Longrightarrow \qquad \gamma_{\text{opt}}(p, d, \Delta) = \frac{p^3}{8 + 2p + (2+\Delta)p^2}$$

Note that this learning rate correspond to an exit time whose log's prefactor is constant to $\frac{1}{4}$, as reported in the main. We can also compute the optimal number of steps; we choose to stick with the *annealed formula*: for large p both estimations lead to the same result, while for small p we are underestimating the exit time of a small factor. Hence, the annealed minimum number of steps is

$$\begin{aligned} s_{\text{ext}}^{\min}(p, d, \Delta, T) \coloneqq & s_{\text{ext}}^{\text{anl}}(p, d, \gamma_{\text{opt}}(p, d, \Delta), \Delta, T) = \\ = & \frac{d\left[8 + 2p + (2+\Delta)p^2\right] \log \left[\frac{T(p+1)d + (p+1)(1-T)}{2p} \right]}{4p^2}. \end{aligned}$$

This formula can be used to estimate the overparameterization gain. Noting that $s_{\text{ext}}^{\min}$ is monotonically decreasing in p, we can define the gain as

$$\lim_{d \to +\infty} \frac{s_{\text{ext}}^{\min}(p = 1, d, \Delta, T)}{\lim_{p \to +\infty} s_{\text{ext}}^{\min}(p, d, \Delta, T)} = \frac{12 + \Delta}{2 + \Delta}.$$

C.4 Stochastic correction to the exit-time formula

In this section we derive a new exit time formula for the case $p = 1$, that takes into account the stochastic corrections of the dynamics.

At first, we require the further assumption $w^0 \perp \boldsymbol{w}_\star^0$. Obviously, this is not realistic, since to achieve this initialization we should know $\boldsymbol{w}_\star$ exactly. Still, this case is could arise interest, since it corresponds to $m^0 = 0$, that is a fixed point of Equation (3.19). Thus, there is no dynamics

in the ODE description, while the SDE are able to jump out from the fixed point and reach the point of convergence of the SGD. Lastly, the following analysis can be generalized to be used with the usual initial conditions.

The key observation is approximating both the noise and the drift of Equation (C.13) to the first non-zero order, when evaluated at $m = 0$. As we pointed out above, the drift term is null at initialization, so the leading order is the first; this corresponds to the linearization of Equation (3.19), and the linear factor is

$$\mu \coloneqq \left[4(1 - 6\gamma) - 2\gamma\Delta\right]. \tag{C.29}$$

Instead, the noise term is not vanishing at $m = 0$. Of course, $\boldsymbol{\sigma}_{\mathrm{Q}}(\Omega)$ does not contribute because is proportional to m_S. Moreover, looking at the expression for Σ in Appendix C.5, we can observe that the covariance matrix is diagonal since the non-diagonal term is proportional to m_S as well. All considered, the only term is contributing is the variance of m: the two-dimensional Brownian motion can be reduced to one in dimension 1, with variance

$$\sigma^2 \coloneqq \frac{\gamma}{pd}(48 + 4\Delta).$$

Summarizing, the SDE near the initialization can be described as

$$\mathrm{d}m = \mu m \,\mathrm{d}t + \sigma \,\mathrm{d}b_t, \tag{C.30}$$

where b_t is a one-dimensional Wiener process with unit variance. The approximated evolution of m is a *expansive Ornstein–Uhlenbeck* process, since $\mu > 0$. We have already shown in the main that there is a direct map between m and the population risk; we can deal just with m for measuring the exit time. Given an expansive OU-process staring at $m = 0$, the mean first exit time from the interval $(-\sqrt{T}, \sqrt{T})$ is given by [Zen20]

$$t_{\text{ext}}^{(SDE)} = \frac{T}{\sigma^2}\,{}_2F_2\left(1, 1; 3/2, 2; -\frac{\mu}{\sigma^2}T\right) \tag{C.31}$$

where ${}_2F_2$ is a generalized hypergeometric function; see [Zen20] for reference. The formula does not have a closed form like the one derived earlier, and it works only when T is small enough that the exit point is not too far from initial conditions, otherwise the approximation we did do not hold anymore.

As reported in [Zen20], the formula can be generalized also to the initial condition we used in the rest of the manuscript, where $\mathrm{M} \sim \mathcal{N}\left(0, 1/d\right)$. Again, we have to average over the initial conditions: (C.31) coincides with the *annealed* version, while we do not present here the *quenched* one to be concise.

C.5 SDEs for arbitrary width

In this appendix we generalize the discussion of the spherical constrained SDE to network of arbitrary width. As we already did at beginning of Section 3.4, we fix the second layer to $a_j = 1$ for everything that follows.

C.5.1 Unconstrained SDE with $p > 1$

The updates of overlapping matrix elements can be written as

$$\begin{aligned}
\mathrm{d}\mathrm{Q}_{jl} &= \frac{-\gamma(\boldsymbol{w}_j \cdot \nabla_{\boldsymbol{w}_l}\ell + \boldsymbol{w}_l \cdot \nabla_{\boldsymbol{w}_j}\ell) + \gamma^2 \nabla_{\boldsymbol{w}_j}\ell^\nu \cdot \nabla_{\boldsymbol{w}_l}\ell^\nu}{d} = \frac{\gamma}{pd}\mathcal{Q}_{jl}, \\
\mathrm{d}m_j &= \frac{-\gamma \boldsymbol{w}_\star \cdot \nabla_{\boldsymbol{w}_j}\ell}{d} = \frac{\gamma}{pd}\mathcal{M}_j.
\end{aligned}$$

where we recalled the definition of the random variables $\mathcal{M}_j$ and $\mathcal{Q}_{jl}$; the factor $1/p$ is missing, but it will come out from the gradients $\nabla_{\boldsymbol{w}_j}$. As we already seen, the usual argument provides setting $\mathrm{d}t = \gamma/pd$ and say that the remaining factor is concentrating to its expected value

$$\begin{aligned}
\mathrm{d}m_j &= \Psi_j \, \mathrm{d}t \\
\mathrm{d}\mathrm{Q}_{jl} &= \Phi_{jl} \, \mathrm{d}t,
\end{aligned}$$

where $\Phi_{jl} = \mathbb{E}\left[\mathcal{Q}_{jl}\right]$ and $\Psi_j = \mathbb{E}\left[\mathcal{M}_j\right]$. We can now go beyond this and add corrections to concentration, namely adding a Brownian motion, following [BGJ22]:

$$\begin{aligned}
\mathrm{d}\mathrm{M}_j &= \Psi_j \, \mathrm{d}t + \sqrt{\frac{\gamma}{pd}}\boldsymbol{\sigma}_j^{\mathrm{M}} \cdot \mathrm{d}\boldsymbol{B}_t \\
\mathrm{d}\mathrm{Q}_{jl} &= \Phi_{jl} \, \mathrm{d}t + \sqrt{\frac{\gamma}{pd}}\boldsymbol{\sigma}_{jl}^{\mathrm{Q}} \cdot \mathrm{d}\boldsymbol{B}_t.
\end{aligned} \tag{C.32}$$

The $\boldsymbol{\sigma}_j^{\mathrm{M}}$ and $\boldsymbol{\sigma}_{jl}^{\mathrm{Q}}$ are the rows of the matrix obtained by taking the square root of the covariance matrix of all the $p + p^2$ random variable $\mathcal{M}_j$ and $\mathcal{Q}_{jl}$:

$$\begin{pmatrix} \boldsymbol{\sigma}_1^{\mathrm{M}} \\ \vdots \\ \boldsymbol{\sigma}_p^{\mathrm{M}} \\ \boldsymbol{\sigma}_{11}^{\mathrm{Q}} \\ \vdots \\ \boldsymbol{\sigma}_{pp}^{\mathrm{Q}} \end{pmatrix} \coloneqq \sqrt{\begin{pmatrix} \mathrm{Var}\left[\mathcal{M}_1\right] & \cdots & \mathrm{Cov}\left[\mathcal{M}_1, \mathcal{M}_p\right] & \mathrm{Cov}\left[\mathcal{M}_1, \mathcal{Q}_{11}\right] & \cdots & \mathrm{Cov}\left[\mathcal{M}_1, \mathcal{Q}_{pp}\right] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}\left[\mathcal{M}_p, \mathcal{M}_1\right] & \cdots & \mathrm{Var}\left[\mathcal{M}_p\right] & \mathrm{Cov}\left[\mathcal{M}_p, \mathcal{Q}_{11}\right] & \cdots & \mathrm{Cov}\left[\mathcal{M}_p, \mathcal{Q}_{pp}\right] \\ \mathrm{Cov}\left[\mathcal{Q}_{11}, \mathcal{M}_1\right] & \cdots & \mathrm{Cov}\left[\mathcal{Q}_{11}, \mathcal{M}_p\right] & \mathrm{Var}\left[\mathcal{Q}_{11}\right] & \cdots & \mathrm{Cov}\left[\mathcal{Q}_{11}, \mathcal{Q}_{pp}\right] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}\left[\mathcal{Q}_{pp}, \mathcal{M}_1\right] & \cdots & \mathrm{Cov}\left[\mathcal{Q}_{pp}, \mathcal{M}_p\right] & \mathrm{Cov}\left[\mathcal{Q}_{pp}, \mathcal{Q}_{11}\right] & \cdots & \mathrm{Var}\left[\mathcal{Q}_{pp}\right] \end{pmatrix}}, \tag{C.33}$$

and $\mathrm{d}B_t$ is a $p(1+p)$-dimensional Wiener process.

C.5.2 Spherical Constrained SDE

Let's move now to the actual spherical derivation. The steps used are exactly the same we did in C.2, Since we are forcing the weight on the sphere, the update rule that has to be used is

$$\boldsymbol{w}_j^{\nu+1} = \frac{\boldsymbol{w}_j^\nu - \gamma \nabla_{\boldsymbol{w}_j}\ell^\nu}{\left\| \boldsymbol{w}_j^\nu - \gamma \nabla_{\boldsymbol{w}_j}\ell^\nu \right\|}\sqrt{d};$$

We introduce now $\mathfrak{M}_j$ and $\mathfrak{Q}_{jl}$ to discriminate between the spherical variable and the unconstrained ones. Multiplying both sides of the update rule by $\boldsymbol{w}_\star$ and subtracting $\mathfrak{M}_j$ we get

$$\mathrm{d}\mathfrak{M}_j = \frac{\mathfrak{M}_j + \mathrm{d}m_{jr}}{\left\|\boldsymbol{w}_j - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|}\sqrt{d} - \mathfrak{M}_j.$$

Similarly, the product of two update rules brings us to

$$\begin{aligned}\mathrm{d}\mathfrak{Q}_{jl} &= \frac{\boldsymbol{w}_j - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu}{\left\|\boldsymbol{w}_j - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|}\frac{\boldsymbol{w}_l - \gamma\nabla_{\boldsymbol{w}_l}\ell^\nu}{\left\|\boldsymbol{w}_l - \gamma\nabla_{\boldsymbol{w}_l}\ell^\nu\right\|} - \mathfrak{Q}_{jl}\\ &= \frac{\mathfrak{Q}_{jl} + \mathrm{dQ}_{jl}}{\left\|\boldsymbol{w}_j - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|\left\|\boldsymbol{w}_l - \gamma\nabla_{\boldsymbol{w}_l}\ell^\nu\right\|}d - \mathfrak{Q}_{jl}\end{aligned}$$

Let's estimate the normalization factor

$$\begin{aligned}\left\|\boldsymbol{w}_j - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu\right\| =& \sqrt{(\boldsymbol{w}_j - \gamma\nabla_{\boldsymbol{w}_j}\ell^\nu)^2} = \sqrt{\boldsymbol{w}_j^2 - 2\boldsymbol{w}_j\cdot\gamma\nabla_{\boldsymbol{w}_j}\ell^\nu + \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2}\\ =& \sqrt{d}\sqrt{\frac{\boldsymbol{w}_j^2}{d} + \frac{-2\boldsymbol{w}_j\cdot\gamma\nabla_{\boldsymbol{w}_j}\ell^\nu + \gamma^2\left\|\nabla_{\boldsymbol{w}_j}\ell^\nu\right\|^2}{d}} = \sqrt{d}\sqrt{\mathrm{Q}_{jj} + \mathrm{dQ}_{jj}}\\ =& \sqrt{d}\sqrt{1 + \mathrm{dQ}_{jj}},\end{aligned}$$

Expanding up to leading orders we get

$$\begin{aligned}\mathrm{d}\mathfrak{M}_j =& (\mathfrak{M}_j + \mathrm{d}m_{jr})(1 + \mathrm{dQ}_{jj})^{-\frac{1}{2}} - \mathfrak{M}_j\\ =& (\mathfrak{M}_j + \mathrm{d}m_{jr})\left(1 - \frac{1}{2}\mathrm{dQ}_{jj} + \frac{3}{8}\mathrm{d}q_{jj}^2\right) - \mathfrak{M}_j\\ =& \mathrm{d}m_{jr} - \frac{\mathfrak{M}_j}{2}\mathrm{dQ}_{jj} - \frac{1}{2}\mathrm{d}m_{jr}\,\mathrm{dQ}_{jj} + \frac{3}{8}\mathfrak{M}_j\,\mathrm{dQ}_{jj}^2\end{aligned}$$

$$\begin{aligned}\mathrm{d}\mathfrak{Q}_{jl} =& (\mathfrak{Q}_{jl} + \mathrm{dQ}_{jl})(1 + \mathrm{dQ}_{jj})^{-\frac{1}{2}}(1 + \mathrm{dQ}_{ll})^{-\frac{1}{2}} - \mathfrak{Q}_{jl}\\ =& (\mathfrak{Q}_{jl} + \mathrm{dQ}_{jl})\left(1 - \frac{1}{2}\mathrm{dQ}_{jj} + \frac{3}{8}\mathrm{d}q_{jj}^2\right)\left(1 - \frac{1}{2}\mathrm{dQ}_{ll} + \frac{3}{8}\mathrm{d}q_{ll}^2\right) - \mathfrak{Q}_{jl}\\ =& \mathrm{dQ}_{jl} - \frac{\mathfrak{Q}_{jl}}{2}(\mathrm{dQ}_{jj} + \mathrm{dQ}_{ll}) + \frac{\mathfrak{Q}_{jl}}{8}(3\,\mathrm{dQ}_{jj}^2 + 3\,\mathrm{dQ}_{ll}^2 + 2\,\mathrm{dQ}_{jj}\,\mathrm{dQ}_{ll}) - \frac{1}{2}(\mathrm{dQ}_{jl}\,\mathrm{dQ}_{jj} + \mathrm{dQ}_{jl}\,\mathrm{dQ}_{ll})\end{aligned}$$

We can now use the Itô Lemma on differentials Equations (C.32), obtaining

$$\mathrm{d}x_a\,\mathrm{d}x_b = \frac{\gamma}{pd}\boldsymbol{\sigma}_{x_a}\cdot\boldsymbol{\sigma}_{x_b}\,\mathrm{d}t,$$

so we have some extra drift terms at an higher order.

C.5.3 Special cases

Computing explicitly the variance in Equation (C.33) is not conceptually different from what exposed in Appendix C.1: expanding all the expression we are left expectations of polynomials of the preactivations. Even if not complex, it is required to compute up to twelfth moments of a multivariate normal distributions, that could lead to very long results. We developed a Mathematica script for addressing this computation, available in the GitHub repository[2]; it can be used for the covariance of arbitrary network width. In the same repository, we provide the code for numerically integrating the SDE, in the cases $p = 1, 2$. These two special case are briefly discussed here.

Explicit expression for $p = 1$ We report here the expression for the variances and covariance introduced in (C.11)

$$\begin{aligned}
\mathrm{Var}\left[\mathcal{Q}_{11}\right] =& 576\gamma^2\Delta\mathrm{M}^4 - 2496\gamma^2\Delta\mathrm{M}^2 - 11520\gamma^2\mathrm{M}^6 + 54144\gamma^2\mathrm{M}^4 - 73728\gamma^2\mathrm{M}^2 \\
&+ 544\gamma\Delta\mathrm{M}^2 - 11136\gamma\mathrm{M}^4 + 22272\gamma\mathrm{M}^2 + 320\mathrm{M}^4 - 1600\mathrm{M}^2 + 32\gamma^2\Delta^2 \\
&+ 1920\gamma^2\Delta + 31104\gamma^2 - 544\gamma\Delta - 11136\gamma + 48\Delta + 1280 \\
\mathrm{Cov}\left[\mathcal{M}_1, \mathcal{Q}_{11}\right] =& 72\gamma\Delta\mathrm{M}^3 - 72\gamma\Delta\mathrm{M} - 1440\gamma\mathrm{M}^5 + 2880\gamma\mathrm{M}^3 \\
&- 1440\gamma\mathrm{M} + 24\Delta\mathrm{M} - 480\mathrm{M}^3 + 480\mathrm{M} \\
\mathrm{Var}\left[\mathcal{M}_1\right] =& 8\Delta\mathrm{M}^2 - 192\mathrm{M}^4 + 144\mathrm{M}^2 + 4\Delta + 48
\end{aligned} \tag{C.34}$$

Numerical experiments for $p = 2$ In Figure C.2 we show the same numerical experiment we presented in Section 3.3.3, repeated in the case $p = 2$. Again, we show there is no benefit including the stochasticity in the analysis. All the evidence indicate that even for larger p there is no effect in including the correction and the time estimation based on the ODE is accurate.

[2] `https://github.com/IdePHICS/EscapingMediocrity`

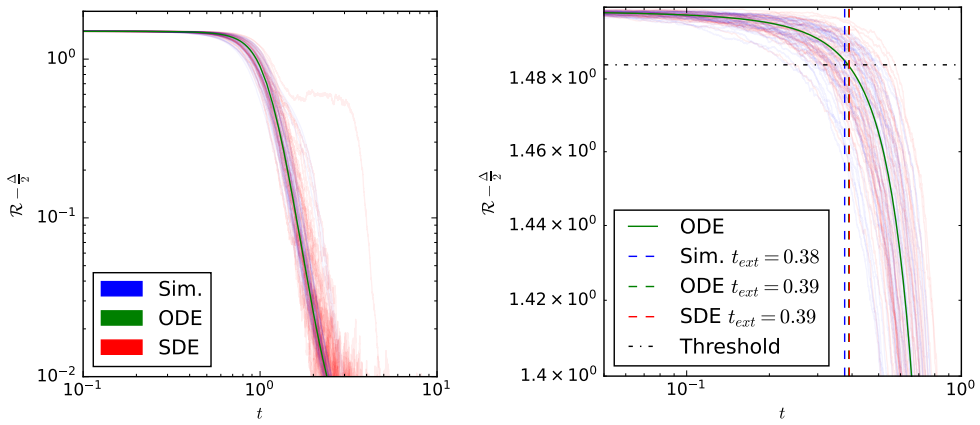


Figure C.2: multiple run of the simulated SGD and the numerically integrated SDE, always starting from the same initial condition, with $d = 3000$. All the t_{ext} presented are obtained by solving numerically (3.21). The SDE captures the variance that the ODE doesn't exhibit, but the t_{ext} do not change considerably.

C.6 Stochastic correction from [BGJ22]

In this Appendix we compute the diffusion correction term of the deterministic dynamic using the formalism introduced in [BGJ22]; we use the notation of that paper to make the comparison easier.

We are going to analyze the plain phase retrieval problem ($k = p = 1$, no spherical constraint), using a slightly different scaling as the one we introduced in the main, in accordance with the hypothesis required by [BGJ22]. Our goal is to show that our formalism contains the one presented in the paper we are analyzing here. We want to learn a predictor of the form $f(x) = (w \cdot x)^2$, using online SGD on the data

$$x \sim \mathcal{N}(0, I_d) \qquad y = (w^\star \cdot x)^2 + \sqrt{\Delta}\xi,$$

with both w and $w^\star$ with $O_d(1)$ norm, in contrast to what we used in the main where the norm of the weights were $O_d(\sqrt{d})$. The SGD process is written as

$$w^{\nu+1} = w^\nu - \delta_d \nabla_w L(y, w),$$

with $\delta_d = \gamma/d$ and $w^0 \sim \mathcal{N}\left(0, {}^1\!/_d I_d\right)$. We are going to consider the high-dimensional limit, namely $d \to +\infty$.

We omit the verification of δ-localizability for brevity.

The loss and its expectation are

$$\begin{aligned}
L(y,w) =& \frac{1}{2}\left[(w^\star\cdot x)^2-(w\cdot x)^2+\sqrt{\Delta}\xi\right]^2\\
=& \frac{1}{2}(w^\star\cdot x)^4+\frac{1}{2}(w\cdot x)^4+\frac{\Delta}{2}\xi^2+\left[(w^\star\cdot x)^2-(w\cdot x)^2\right]\sqrt{\Delta}\xi-\left((w^\star\cdot x)(w\cdot x)\right)^2,\\
\Phi(w) \coloneqq& \mathbb{E}_{x,\xi}\left[L(y,w)\right]=\frac{3}{2}\left\|w^\star\right\|^4+\frac{3}{2}\|w\|^4+\frac{3}{2}\Delta-\left\|w^\star\right\|^2\|w\|^2-2(w^\star\cdot w)^2
\end{aligned}$$

We introduce our summary statistics $u=(m,q)\coloneqq(w^\star\cdot w,\|w\|^2)$, and we further assume $\|w^\star\|^2=\rho$:

$$\begin{aligned}
\Phi(w) =& \frac{3\|w^\star\|^4+\Delta}{2}+\frac{3}{2}\|w\|^4-\left\|w^\star\right\|^2\|w\|^2-2(w^\star\cdot w)^2\\
\Phi(m,q) =& \frac{3\rho^2+\Delta}{2}+\frac{3}{2}q^2-\rho q-2m^2
\end{aligned}$$

We now compute the gradients needed for the dynamics. The stochastic gradient is

$$\nabla_w L=2(w\cdot x)^3x-2(w\cdot x)\sqrt{\Delta}\xi x-2(w^\star\cdot x)^2(w\cdot x)x$$

The gradient of the population loss is

$$\nabla_w\Phi=6qw-2\rho w-4mw^\star$$

The gradients of the sufficient statistics with respect to w are

$$\nabla_w m=w^\star\qquad\nabla_w q=2w,$$

and their Hessians are

$$\nabla_w^2 m=0\qquad\nabla_w^2 q=2I_d.$$

Last ingredient we need for the dynamic equations is $V\coloneqq\mathbb{E}_{x,\xi}\left[\nabla_w(L-\Phi)\otimes\nabla_w(L-\Phi)\right]$.

We are left with

$$\begin{aligned}
V=&\mathbb{E}_{x,\xi}\left[\nabla_w L\otimes\nabla_w L\right]+\nabla_w\Phi\otimes\nabla_w\Phi\\
&-\mathbb{E}_{x,\xi}\left[\nabla_w L\otimes\nabla_w\Phi+\nabla_w\Phi\otimes\nabla_w L\right].
\end{aligned}$$

We compute the two terms separately

$$\begin{aligned}
\nabla_w\Phi\otimes\nabla_w\Phi=&4(3q-\rho)^2ww^\top\\
&-4(6qm-2\rho m)(w^\star w^\top+ww^{\star\top})+4\cdot4m^2w^\star w^{\star\top}
\end{aligned}$$

Since $\mathbb{E}_{x,\xi}\left[\nabla_w L\right]=\nabla_w\Phi$, the cross term reduces to the same quantity:

$$\begin{aligned}
\mathbb{E}_{x,\xi}\left[\nabla_w L\right]\otimes\nabla_w\Phi=&(6\|w\|^2w-4(w^\star\cdot w)w^\star-2\left\|w^\star\right\|w)\otimes(6qw-2\rho w-4mw^\star)\\
=&\nabla_w\Phi\otimes\nabla_w\Phi.
\end{aligned}$$

It remains to compute the second-moment term:

$$
\begin{aligned}
&\mathbb{E}_{x,\xi}\left[\nabla_w L \otimes \nabla_w L\right] = \\
&\quad = \mathbb{E}_{x,\xi}\Big[4(w\cdot x)^6 xx^\top + 4(w\cdot x)^2\Delta\xi^2 xx^\top \\
&\qquad\qquad + 4(w^\star\cdot x)^4(w\cdot x)^2 xx^\top - 8(w\cdot x)^4(w^\star\cdot x)^2 xx^\top\Big] \\
&\quad = 4(90\|w\|^4 ww^\top + 15\|w\|^6 I_d) + 4\Delta(\|w\|^2 I_d + 2ww^\top) \\
&\qquad + 4\big[(3\left\|w^\star\right\|^4\|w\|^2 + 12\left\|w^\star\right\|^2(w^\star\cdot w)^2)I_d \\
&\qquad\quad + 6\left\|w^\star\right\|^4 ww^\top + (12\left\|w^\star\right\|^2\|w\|^2 + 24(w^\star\cdot w)^2)w^\star(w^\star)^\top \\
&\qquad\quad + 24\left\|w^\star\right\|^2(w^\star\cdot w)(w^\star w + ww^\star)\big] \\
&\qquad - 8\big[(3\|w\|^4\left\|w^\star\right\|^2 + 12\|w\|^2(w^\star\cdot w)^2)I_d \\
&\qquad\quad + 6\|w\|^4 w^\star(w^\star)^\top + (12\|w\|^2\left\|w^\star\right\|^2 + 24(w^\star\cdot w)^2)ww^\top \\
&\qquad\quad + 24\|w\|^2(w^\star\cdot w)(w^\star w + ww^\star)\big] \\
&\quad = 4(15q^3 + \Delta q + 3\rho^2 q + 12\rho m^2 - 6q^2\rho - 24qm^2)I_d \\
&\qquad + 4(90q^2 + 2\Delta + 6\rho^2 - 24\rho q - 48m^2)ww^\top \\
&\qquad + 4(12\rho q + 24m^2 - 12q^2)w^\star(w^\star)^\top \\
&\qquad + 4(24\rho m - 48qm)(w^\star w + ww^\star)
\end{aligned}
$$

We finally get the expression for the matrix $V = \mathbb{E}_{x,\xi}\left[\nabla_w L \otimes \nabla_w L\right] - \nabla_w\Phi \otimes \nabla_w\Phi$

$$
\begin{aligned}
V = &\ 4(15q^3 + \Delta q + 3\rho^2 q + 12\rho m^2 - 6q^2\rho - 24qm^2)I_d \\
&+ 4(90q^2 + 2\Delta + 6\rho^2 - 24\rho q - 48m^2 - 9q^2 + 6\rho q - \rho^2)ww^\top \\
&+ 4(12\rho q + 24m^2 - 12q^2 - 4m^2)w^\star(w^\star)^\top \\
&+ 4(24\rho m - 48qm + 6qm - 2\rho m)(w^\star w + ww^\star)
\end{aligned}
\tag{C.35}
$$

We now have all the ingredients needed for computing the dynamics

$$
\begin{aligned}
f_m &= -\lim_{d\to+\infty} \nabla_w\Phi\cdot\nabla_w m = -6qm + 2\rho m + 4\rho m = 6m(\rho - q) \\
f_q &= -\lim_{d\to+\infty} \nabla_w\Phi\cdot\nabla_w q = -12q^2 + 4\rho q + 8m^2 \\
g_m &= \lim_{d\to+\infty} \frac{\delta_d}{2}\operatorname{Tr}\left[V\nabla_w^2 m\right] = 0 \\
g_q &= \lim_{d\to+\infty} \frac{\delta_d}{2}\operatorname{Tr}\left[V\nabla_w^2 q\right] \\
&= 4(15q^3 + \Delta q + 3\rho^2 q + 12\rho m^2 - 6q^2\rho - 24qm^2)\cdot\lim_{d\to+\infty}\frac{\gamma}{2d}\operatorname{Tr}\left[2I_d\right] \\
&= 4\gamma(15q^3 + \Delta q + 3\rho^2 q + 12\rho m^2 - 6q^2\rho - 24qm^2)
\end{aligned}
$$

In the last one we used the fact that the only term that can survive the $\mathrm{Tr}\left[\cdot\right]/d$ are the ones proportional to the identity. For the same reason $\Sigma = 0$ in this case. The full equations read as

$$\begin{aligned} \mathrm{d}m &= 6m(\rho - q)\,\mathrm{d}t \\ \mathrm{d}q &= 4\Big[2m^2 + \rho q - 3q^2 \\ &\qquad + \gamma(15q^3 + \Delta q + 3\rho^2 q + 12\rho m^2 - 6q^2\rho - 24qm^2)\Big]\,\mathrm{d}t \end{aligned} \tag{C.36}$$

Note that this equations are exactly the same as (C.5a) and (C.5b), as expected. What is not trivial is proving that the diffusion term will also be equivalent.

C.6.1 Diffusion at generic point

In principle diffusion terms are expected to appear when the sufficient statistic are zoom around a fixed point of the equations. Imagine now we want to find the diffusion term around a generic point $(m, q) = (\bar{m}, \bar{q})$ that is not necessarily a fixed point of the equations. We have to zoom the sufficient statistics in the usual way

$$(\tilde{m}, \tilde{q}) = \left(\sqrt{d}(m - \bar{m}), \sqrt{d}(q - \bar{q})\right)$$

. The drift terms diverge, but diffusion term will appear; this diffusion is the first stochastic correction to the deterministic dynamic of the sufficient statistics, and can be added to the equations (C.36), paying attention to multiply it by $1/\sqrt{d}$ to compensate the zoom factor.

We express the loss and its gradients in terms of the new sufficient statistics.

$$\Phi(\tilde{m}, \tilde{q}) = \frac{3\rho^2 + \Delta}{2} + \frac{3}{2}\left(\frac{\tilde{q}}{\sqrt{d}} + \bar{q}\right)^2 - \rho\left(\frac{\tilde{q}}{\sqrt{d}} + \bar{q}\right) - 2\left(\frac{\tilde{m}}{\sqrt{d}} + \bar{m}\right)^2$$

Its gradient with respect to w is

$$\nabla_w \Phi = 6\left(\frac{\tilde{q}}{\sqrt{d}} + \bar{q}\right) w - 2\rho w - 4\left(\frac{\tilde{m}}{\sqrt{d}} + \bar{m}\right) w^\star.$$

The gradients of the zoomed statistics are

$$\nabla_w \tilde{m} = \sqrt{d} w^\star \qquad \nabla_w \tilde{q} = 2\sqrt{d} w,$$

and their Hessians are

$$\nabla_w^2 \tilde{m} = 0 \qquad \nabla_w^2 \tilde{q} = 2\sqrt{d} I_d.$$

The expression of matrix V is the same as the one in equation (C.35) with the following substitutions

$$q = \frac{\tilde{q}}{\sqrt{d}} + \bar{q} \qquad m = \frac{\tilde{m}}{\sqrt{d}} + \bar{m}.$$

We now compute the three entries of the diffusion matrix in turn.

$$
\begin{aligned}
\Sigma_{\tilde{m},\tilde{m}} &= \lim_{d\to+\infty} \delta_d(\nabla_w \tilde{m})^\top V \nabla_w \tilde{m} \\
&= \lim_{d\to+\infty} \Big[4(15q^3+\Delta q+3\rho^2 q+12\rho m^2-6q^2\rho-24qm^2)\rho \\
&\qquad +4(90q^2+2\Delta+6\rho^2-24\rho q-48m^2-9q^2+6\rho q-\rho^2)m^2 \\
&\qquad +4(12\rho q+24m^2-12q^2-4m^2)\rho^2 \\
&\qquad +8(24\rho m-48qm+6qm-2\rho m)\rho m\Big] \\
&=60\rho\bar{q}^3+4\Delta\rho\bar{q}+12\rho^3\bar{q}+48\rho^2\bar{m}^2-24\rho^2\bar{q}^2-96\rho\bar{q}\bar{m}^2 \\
&\quad 324\bar{m}^2\bar{q}^2+8\Delta\bar{m}^2+24\rho^2\bar{m}^2-96\rho\bar{q}\bar{m}^2-192\bar{m}^4+24\rho\bar{q}\bar{m}^2-4\rho^2\bar{m}^2 \\
&\quad +48\rho^3\bar{q}+96\rho^2\bar{m}^2-48\rho^2\bar{q}^2-16\rho^2\bar{m}^2 \\
&\quad +176\rho^2\bar{m}^2-336\rho\bar{q}\bar{m}^2 \\
&=60\rho\bar{q}^3+4\Delta\rho\bar{q}+60\rho^3\bar{q}+324\rho^2\bar{m}^2-72\rho^2\bar{q}^2-504\rho\bar{q}\bar{m}^2 \\
&\quad +8\Delta\bar{m}^2-192\bar{m}^4+324\bar{m}^2\bar{q}^2
\end{aligned}
$$

The off-diagonal entry is

$$
\begin{aligned}
\Sigma_{\tilde{m},\tilde{q}} &= \lim_{d\to+\infty} \delta_d(\nabla_w \tilde{q})^\top V \nabla_w \tilde{m} \\
&= \lim_{d\to+\infty} \Big[(15q^3+\Delta q+3\rho^2 q+12\rho m^2-6q^2\rho-24qm^2)m \\
&\qquad +(90q^2+2\Delta+6\rho^2-24\rho q-48m^2-9q^2+6\rho q-\rho^2)qm \\
&\qquad +(12\rho q+24m^2-12q^2-4m^2)m\rho \\
&\qquad +(24\rho m-48qm+6qm-2\rho m)(m^2+\rho q)\Big] \\
&=120\bar{m}\bar{q}^3+8\Delta\bar{m}\bar{q}+24\rho^2\bar{m}\bar{q}+96\rho\bar{m}^3-48\rho\bar{m}\bar{q}^2-192\bar{m}^3\bar{q} \\
&\quad 720\bar{m}\bar{q}^3+16\Delta\bar{m}\bar{q}+48\rho^2\bar{m}\bar{q}-192\rho\bar{m}\bar{q}^2-384\bar{m}^3\bar{q}-72\bar{m}\bar{q}^3+48\rho\bar{m}\bar{q}^2-8\rho^2\bar{m}\bar{q} \\
&\quad 96\rho^2\bar{m}\bar{q}+192\rho\bar{m}^3-96\rho\bar{m}\bar{q}^2-32\rho\bar{m}^3 \\
&\quad 192\rho\bar{m}^3-384\bar{m}^3\bar{q}+48\bar{m}^3\bar{q}-16\rho\bar{m}^3 \\
&\quad 192\rho^2\bar{m}\bar{q}-384\rho\bar{m}\bar{q}^2+48\rho\bar{m}\bar{q}^2-16\rho^2\bar{m}\bar{q} \\
&=768\bar{m}\bar{q}^3+24\Delta\bar{m}\bar{q}+336\rho^2\bar{m}\bar{q}+432\rho\bar{m}^3-624\rho\bar{m}\bar{q}^2-912\bar{m}^3\bar{q}
\end{aligned}
$$

Finally, the bottom-right entry gives

$$
\begin{aligned}
\Sigma_{\tilde{q},\tilde{q}} &= \lim_{d\to+\infty} \delta_d(\nabla_w \tilde{q})^\top V \nabla_w \tilde{q} \\
&= \lim_{d\to+\infty} 16\Big[(15q^3 + \Delta q + 3\rho^2 q + 12\rho m^2 - 6q^2\rho - 24qm^2)q \\
&\qquad + (90q^2 + 2\Delta + 6\rho^2 - 24\rho q - 48m^2 - 9q^2 + 6\rho q - \rho^2)q^2 \\
&\qquad + (12\rho q + 24m^2 - 12q^2 - 4m^2)m^2 \\
&\qquad + 2(24\rho m - 48qm + 6qm - 2\rho m)qm\Big] \\
&= 240\bar{q}^4 + 16\Delta\bar{q}^2 + 48\rho^2\bar{q}^2 + 192\rho\bar{m}^2\bar{q} - 96\rho\bar{q}^3 - 384\bar{q}^2\bar{m}^2 \\
&\quad 1440\bar{q}^4 + 32\Delta\bar{q}^2 + 96\rho^2\bar{q}^2 - 384\rho\bar{q}^3 - 768\bar{m}^2\bar{q}^2 - 144\bar{q}^4 + 96\rho\bar{q}^3 - 16\rho^2\bar{q}^2 \\
&\quad 192\rho\bar{m}^2\bar{q} + 384\bar{m}^4 - 192\bar{m}^2\bar{q}^2 - 64\bar{m}^4 \\
&\quad 768\rho\bar{m}^2\bar{q} - 1536\bar{m}^2\bar{q}^2 + 192\bar{m}^2\bar{q}^2 - 64\rho\bar{m}^2\bar{q} \\
&= 1536\bar{q}^4 + 48\Delta\bar{q}^2 + 128\rho^2\bar{q}^2 + 1088\rho\bar{m}^2\bar{q} - 384\rho\bar{q}^3 - 2688\bar{q}^2\bar{m}^2 + 320\bar{m}^4
\end{aligned}
$$

The expressions we just computed match the one we obtained in Equation (C.33), if we set $\gamma = 0$. This difference arise from the δ-localizability condition, that essentially force us to work with a vanishing learning rate. Remark 2 in [BGJ22] explains how to generalize the formula for Σ to match exactly our expression.

C.7 Landscape geometry

In this appendix we discuss the landscape of the population risk for the phase retrieval problem $p = 1$. We start by discussing the Euclidean case first, moving to the sphere next.

We recall the reader that the loss function is given by given by:

$$
\ell(w) \coloneqq \ell(y, f_\Theta(x)) = \frac{1}{2}\left((w_\star^\top x)^2 + \sqrt{\Delta}z - (w^\top x)^2\right)^2 \tag{C.37}
$$

As before, we define the pre-activations (or local fields):

$$
\lambda_\star = w_\star^\top x, \qquad\qquad \lambda = w^\top x \tag{C.38}
$$

and the displacement vector:

$$
\begin{aligned}
\mathcal{E}(w) &\coloneqq (w_\star^\top x)^2 + \sqrt{\Delta}z - (w^\top x)^2 \\
&= \lambda_\star^2 + \sqrt{\Delta}z - \lambda^2
\end{aligned} \tag{C.39}
$$

Note that:

$$
\nabla_w \mathcal{E}(w) = -2\lambda x \tag{C.40}
$$

Therefore, the Euclidean gradient of the loss is given by:

$$
\nabla_w \ell(w) = -2\mathcal{E}(w)\lambda x \tag{C.41}
$$

And the Euclidean Hessian is:

$$\begin{aligned}\nabla^2\ell(w) &= 2\left(2\lambda^2 - \mathcal{E}(w)\right)xx^\top \\ &= 2(3\lambda^2 - \lambda_\star^2 + \sqrt{\Delta}z)xx^\top\end{aligned} \tag{C.42}$$

Averaged geometry — We now compute the expected geometry by taking population averages of the above. It will be useful to define the correlation variables:

$$\rho = \frac{||w_\star||^2}{d}, \qquad m = \frac{w_\star^\top w}{d}, \qquad q = \frac{||w||^2}{d} \tag{C.43}$$

which we recall are the second moments of the pre-activations $(\lambda_\star, \lambda)$. With this notation, the population risk (the expected value of (C.37)) reads:

$$\mathcal{R}(m,q) = \Delta/2 + \frac{3}{2}\rho^2 + \frac{3}{2}q^2 - 2m^2 - \rho q \tag{C.44}$$

In order to compute the expected gradient, we need the following moments (for readability we omit throughout the overall factor $1/d$ coming from the covariance $\Sigma = 1/dI$, which rescales the gradient and Hessian but changes neither the critical points nor the signs of the Hessian eigenvalues):

$$\mathbb{E}[\lambda_\star^2\lambda x] = \rho w + 2mw_\star, \qquad \mathbb{E}[\lambda^3 x] = 3qw \tag{C.45}$$

Which gives:

$$\nabla_w\mathcal{R}(w) = -2\mathbb{E}\left[\mathcal{E}(w)\lambda x\right] = -2((\rho - 3q)w + 2mw_\star) \tag{C.46}$$

Finally, let's compute the expected Hessian. For that, we will need the following moments:

$$\mathbb{E}[\lambda_\star^2 xx^\top] = \rho I_d + w_\star w_\star^\top, \qquad \mathbb{E}[\lambda^2 xx^\top] = qI_d + ww^\top \tag{C.47}$$

Therefore:

$$\nabla_w^2\mathcal{R}(w) = 2\mathbb{E}\left[(3\lambda^2 - \lambda_\star^2 + \sqrt{\Delta}z)xx^\top\right] = 2\left(-(\rho - 3q)I_d + 3ww^\top - w_\star w_\star^\top\right) \tag{C.48}$$

We are now ready to compute the critical points of the Euclidean landscape and evaluate their nature. By definition, the critical points are defined as solutions of $\nabla_w\mathcal{R}(w) = 0$. These are:

- $w = 0$ ($(m,q) = (0,0)$): The Hessian of this critical point is given by:

  $$\nabla_w^2\mathcal{R}(0) = -2(\rho I_d + w_\star w_\star^\top) \prec 0 \tag{C.49}$$

  This is a negative-definite matrix with $d-1$ negative eigenvalues -2ρ and one negative eigenvalue $-2(\rho+1)$ with eigenvector $w_\star$. Therefore, this is a local maximum. The risk associated is given by:

  $$\mathcal{R}(0) = \frac{\Delta}{2} + \frac{3}{2}\rho^2 \tag{C.50}$$

- $(m, q) = (0, \rho/3)$: This defines a line of critical points $\{w \in \mathbb{R}^d : w \perp w_\star \text{ and } ||w|| = 1/\sqrt{3}||w_\star||\}$. The Hessian is given by:

$$\nabla_w^2 \mathcal{R} = 2(3ww^\top - w_\star w_\star^\top) \tag{C.51}$$

  Note this is a rank-two matrix with $d-2$ zero eigenvalues (associated to flat directions), one negative eigenvalue with eigenvector $w_\star$ and a positive eigenvalue with eigenvector perpendicular to the minima. This is a saddle-point, and have population risk:

$$\mathcal{R}(0, \rho/3) = \frac{\Delta}{2} + \frac{4}{3}\rho^2 \tag{C.52}$$

- $w = \pm w_\star$ ($(m, q) = (\pm\rho, \rho)$): From the definition of our problem, this is the global minima. The expected Hessian is given by:

$$\nabla_w^2 \mathcal{R}(\pm w_\star) = 4(\rho I_d + w_\star w_\star^\top) \succ 0 \tag{C.53}$$

  which is indeed a positive definite matrix. This defines the minimum achievable population risk:

$$\min_{w \in \mathbb{R}^d} \mathcal{R}(w) = \mathcal{R}(\pm\rho, \rho) = \frac{\Delta}{2} \tag{C.54}$$

This is consistent with the discussion in [Che+19], where the critical points and their nature were reported, but explicit expressions for the expected gradient and Hessian were not given. From this geometry, a neat picture for the one-pass SGD dynamics in the unconstrained problem can be drawn. Consider $w_\star \in \mathbb{S}^{d-1}$ with a random initialization at high-dimensions:

$$w^0 \sim \mathcal{N}(0, I_d). \tag{C.55}$$

Note that with high-probability the random initial weights is almost orthogonal to the signal, and we have $(m, q) \approx (0, 1)$. Note this is not a critical point, and the initial gradient $\nabla_w \mathcal{R}(w^0) = 4w^0$ is orthogonal to $w_\star$. Indeed, for $p = 1$ and $\Delta = 0$ the unconstrained ODEs (3.11) are given by:

$$\dot{m}(t) = 6\, m(t)(\rho - q(t)) \tag{C.56}$$

$$\begin{aligned}\dot{q}(t) = 4\Big(q(t)(\rho - 3q(t)) + 2m(t)^2\Big) \\ + 12\gamma\Big(q(t)(\rho^2 + 5q(t)^2 - 2\rho q(t)) + 4m^2(\rho - 2q(t))\Big)\end{aligned} \tag{C.57}$$

where $\dot{} := {}^{\mathrm{d}}/_{\mathrm{d}t}$ and we dropped the bars for clarity. Therefore, in the initial stage of the dynamics $\dot{m} \approx 0$ remains almost constant, while q decreases, and the dynamics flow in the direction of the saddle-point $(0, \rho/3)$. As we have seen, the saddle is mostly flat, with a single negative curvature direction pointing towards $w_\star$. This is precisely the mediocrity stage, where the dynamics slows down and SGD gets stuck for a long time before being able to develop significant correlation with $w_\star$ and escape.

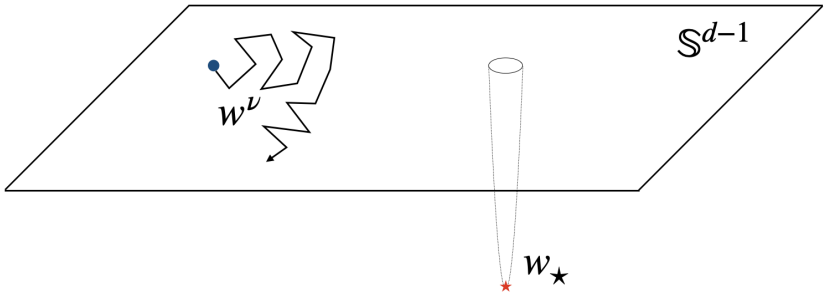


Figure C.3: Low-dimensional illustration of mediocrity at initialization. As discussed in Section C.7.1, the expected Hessian at initialization is a strict saddle-point with $d-1$ flat directions and a single negative direction pointing towards the global minimum. This scenario is particularly hard for descent based algorithms such as SGD, that require $n = d \log d$ samples / steps to develop significant correlation with the signal.

C.7.1 Landscape in the sphere

Recall that the orthogonal projector on a vector $u \in \mathbb{S}^{d-1}$ is given by:

$$\text{Proj}_u = I_d - uu^\top \tag{C.58}$$

Therefore, the gradient on the sphere is given by:

$$\begin{aligned} \text{grad}_{\mathbb{S}^{d-1}} \ell(w) &= \text{Proj}_{\mathbb{S}^{d-1}}(\nabla_w \ell(w)) = (I_d - ww^\top)\nabla_w \ell(w) \\ &= -2\mathcal{E}(w)\lambda(x - \lambda w) \end{aligned} \tag{C.59}$$

Similarly, the Hessian on the sphere can be written as:

$$\begin{aligned} \text{Hess}_{\mathcal{S}^{d-1}} \ell(w) &= \text{Proj}_{\mathbb{S}^{d-1}} \left(\nabla_w^2 \ell(w) \right) - \langle w, \nabla_w \ell(w) \rangle I_d \\ &= (3\lambda^2 - \lambda_\star^2)(xx^\top - \lambda w x^\top) + \mathcal{E}(w)\lambda^2 I_d \end{aligned} \tag{C.60}$$

Averaged geometry — Recall that in the sphere we have $\rho = q = 1$. Therefore, the population risk now only depends on the correlation $m = \langle w_\star, w \rangle$, and reads:

$$\mathcal{R}(w) = 2(1 - m^2) + \frac{\Delta}{2} \tag{C.61}$$

Luckily, half of the moments we need have been already computed above. To get the gradient on the sphere, we just need to compute:

$$\mathbb{E}[\lambda_\star^2 \lambda^2] = 1 + 2m^2, \qquad \mathbb{E}[\lambda^4] = 3 \tag{C.62}$$

Therefore, the averaged spherical gradient is given by:

$$\text{grad}_{\mathbb{S}^{d-1}}\mathcal{R}(w) = 4m\left(mw - w_\star\right) \tag{C.63}$$

We also have all moments required to compute the expected spherical Hessian:

$$\text{Hess}_{\mathcal{S}^{d-1}}\mathcal{R}(w) = 4\left[m^2 I_d + mww_\star^\top - w_\star w_\star^\top\right] \tag{C.64}$$

As before, we can now analyze the critical points and their nature.

- $w \perp w_\star$ ($m = 0$): The Hessian is given by:

$$\text{Hess}_{\mathcal{S}^{d-1}}\mathcal{R}(w) = -4w_\star w_\star^\top \tag{C.65}$$

  Which is a rank one matrix with $d-1$ eigenvalues 0 (flat directions) and a single negative eigenvalue -4 with negative curvature pointing towards the signal $w_\star$. Therefore, this is a strict saddle-point. However, since the risk is a decreasing function of $m^2 \in [0,1]$, this is also the global maximum of the risk.

- $w = \pm w_\star$ ($m = \pm 1$): As before, these are the global minima, and define the minimal achievable risk $\mathcal{R}(\pm w_\star) = \Delta/2$. Indeed, the Hessian is given by:

$$\text{Hess}_{\mathcal{S}^{d-1}}\mathcal{R}(\pm w_\star) = 4I_d \succ 0 \tag{C.66}$$

  which is positive-define.

Therefore, the landscape now resembles a golf course: completely flat with a single whole corresponding with the global minimum $w_\star$, see Figure C.3 for an illustration. This is the prototypical image of mediocrity. In particular, differently from the unconstrained case, random initialization

$$w^0 \sim \text{Unif}(\mathbb{S}^{d-1}(\sqrt{d})) \tag{C.67}$$

now corresponds to initializing close to the saddle point $m^0 = 0$: with high-probability the initial weights are orthogonal to the signal at high-dimensions:

$$m^0 = 1/d\langle w^0, w_\star\rangle \approx 1/\sqrt{d} \ll 1. \tag{C.68}$$

For the reader convenience, we recall that the ODEs (3.19) describing the evolution of the correlation m is given by:

$$\dot{m}(t) = m(t)\left[4(1-6\gamma)(1-m^2(t)) - 2\gamma\Delta\right] \tag{C.69}$$

Close to initialization we now have $\dot{m}(0) \approx 0$, slowing down the dynamics close to initialization.

Appendix D

Appendices to "Online Learning and Information Exponents: On The Importance of Batch size, and Time / Complexity Tradeoffs"

D.1 Proof of Theorems 4.1 and 4.2

D.1.1 Preliminaries

Notations and definitions We denote by polylog x any polynomial in $\log x$ with degree > 1. Since the case where $n_b = O(1)$ is already covered by the results in [BGJ21], we shall always assume that $\mu > 0$. The Hermite coefficients of σ and $f^\star$ will be denoted by $(c_k)_{k\geq 0}$ and $(c_k^\star)_{k\geq 0}$, respectively. To break the symmetry between m and $-m$ inherent to the problem, we assume without loss of generality that

$$m_0 > 0 \quad \text{and} \quad c_\ell c_\ell^\star > 0.$$

Throughout this section, the update process on $\boldsymbol{w}_t$ will be written as

$$\boldsymbol{w}_{t+1} = \frac{\boldsymbol{w}_t - \gamma \boldsymbol{g}_t}{\|\boldsymbol{w}_t - \gamma \boldsymbol{g}_t\|}, \tag{D.1}$$

where $\boldsymbol{g}_t$ is the gradient at time t: $\boldsymbol{g}_t = \nabla_{w_t} \hat{\mathcal{R}}_t$, and $\hat{\mathcal{R}}_t$ is the empirical loss at time t, that can be either the correlation or the square loss. When considering the update of the process $(\boldsymbol{w}_t)_{t\geq 0}$, it will be useful to distinguish between the randomness in $\boldsymbol{w}_t$ and the one introduced by the batch drawn at time t. To this end, we introduce the filtration $(\mathcal{F}_t)_{t\geq 0}$ adapted to the process $\boldsymbol{w}_t$, and we shall denote by $\mathbb{P}_t$ (resp. $\mathbb{E}_t$) the probability (resp. expectation) conditioned on $\mathcal{F}_t$.

Concentration in Orlicz spaces We first recall some fact about Orlicz spaces that will be useful for our concentration bounds.

Definition D.1. *For any $\alpha \in \mathbb{R}$, let $\psi_\alpha(x) = e^{x^\alpha} - 1$. Let X be a real random variable; the* Orlicz norm $\|X\|_{\psi_\alpha}$ *is defined as*

$$\|X\|_{\psi_\alpha} = \inf\left\{ t > 0 : \mathbb{E}\left[\psi_\alpha\left(\frac{|X|}{t}\right)\right] \leq 1 \right\} \tag{D.2}$$

It can be checked that $\|\cdot\|_{\psi_\alpha}$ is a well-defined norm on random variables for $\alpha \geq 1$, and can be slightly modified into a norm when $\alpha < 1$; see [LT91; VW96] for more information. We say that a random variable is sub-gaussian (resp. sub-exponential) if its ψ_2 (resp. ψ_1) norm is finite. The main use of this definition is the following concentration inequality:

Lemma D.1. *Let X be a random variable with finite ψ_α-norm for some $\alpha > 0$. Then*

$$\mathbb{P}\left[|X - \mathbb{E}X| > t\|X\|_{\psi_\alpha}\right] \leq 2e^{-t^\alpha}. \tag{D.3}$$

As a result, any ψ_α-norm bound yields exponential convergence tails. Orlicz norms are also well-behaved with respect to products:

Lemma D.2. *Let X and Y be two random variables such that $\|X\|_{\psi_\alpha}$ and $\|Y\|_{\psi_\beta}$ are finite for some $\alpha, \beta > 0$. Then*

$$\|XY\|_{\psi_\lambda} \leq \|X\|_{\psi_\alpha}\|Y\|_{\psi_\beta},$$

where λ is the number satisfying $\frac{1}{\alpha} + \frac{1}{\beta} = \frac{1}{\lambda}$.

Proof. Assume without loss of generality that $\|X\|_{\psi_\alpha} = \|Y\|_{\psi_\beta} = 1$. We use the following Young inequality: for any $a, b > 0$, and p, q such that $\frac{1}{p} + \frac{1}{q} = 1$,

$$ab \leq \frac{a^p}{p} + \frac{b^q}{q}$$

Applying this inequality to $p = \alpha/\lambda$, $q = \beta/\lambda$, $a = X^\lambda$, $b = Y^\lambda$, we get

$$(XY)^\lambda \leq \frac{\lambda X^\alpha}{\alpha} + \frac{\lambda Y^\beta}{\beta}.$$

Then

$$\begin{aligned}\exp((XY)^\lambda) &\leq \exp\left(\frac{\lambda X^\alpha}{\alpha}\right) + \exp\left(\frac{\lambda Y^\beta}{\beta}\right) \\ &\leq \frac{\lambda}{\alpha}\exp(X^\alpha) + \frac{\lambda}{\beta}\exp(Y^\beta),\end{aligned}$$

where at the last line we used Young's inequality again with the same p and q. The result ensues from taking expectations on both sides, and noticing that $\lambda/\alpha + \lambda/\beta = 1$ by definition. □

Finally, we shall use the following theorem:

Theorem D.1 (Theorem 6.2.3 in [LT91]). *Let $X_1, \dots, X_n$ be n independent random variables with zero mean and second moment $\mathbb{E}X_i^2 = \sigma_i^2$. Then,*

$$\left\| \sum_{i=1}^{n} X_i \right\|_{\psi_\alpha} \leq K_\alpha \log(n)^{1/\alpha} \left(\sqrt{\sum_{i=1}^{n} \sigma_i^2} + \max_i \|X_i\|_{\psi_\alpha} \right) \tag{D.4}$$

D.1.2 Computing the gradient at time t

A direct computation of both gradients implies the following lemma:

Lemma D.3. *Define*

$$\begin{aligned} \boldsymbol{g}_t^\star &= \frac{1}{n_b} \sum_{\nu=1}^{n_b} f^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}^\nu \rangle) \sigma'(\langle \boldsymbol{w}_t, \boldsymbol{z}^\nu \rangle) \boldsymbol{z}^\nu, \\ \hat{\boldsymbol{g}}_t &= \frac{1}{n_b} \sum_{\nu=1}^{n_b} \sigma(\langle \boldsymbol{w}_t, \boldsymbol{z}^\nu \rangle) \sigma'(\langle \boldsymbol{w}_t, \boldsymbol{z}^\nu \rangle) \boldsymbol{z}^\nu, \end{aligned} \tag{D.5}$$

Then the gradient of the correlation loss ℓ^{corr} is $-\boldsymbol{g}_t^\star$, while the gradient of the square loss ℓ^{sq} is $\hat{\boldsymbol{g}}_t - \boldsymbol{g}_t^\star$.

Hence, the main difference between the gradients of the correlation and square loss is a so-called *self-interaction term* $\hat{\boldsymbol{g}}$, that only depends on the learned vector $\boldsymbol{w}_t$. Notice that $\boldsymbol{g}_t^\star$ is an average of n_b independent variables of the form

$$\boldsymbol{g}_t^{\star\nu} := f^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}^\nu \rangle) \sigma'(\langle \boldsymbol{w}_t, \boldsymbol{z}^\nu \rangle) \boldsymbol{z}^\nu, \tag{D.6}$$

and we define $\hat{\boldsymbol{g}}_t^\nu$ and $\boldsymbol{g}_t^\nu$ in the same way. By Assumption 4.1 and Lemma D.2, each variable $\boldsymbol{g}_t^{\star\nu}$ (resp. $\hat{\boldsymbol{g}}_t^\nu, \boldsymbol{g}_t^\nu$) has finite ψ_α-norm for some $\alpha > 0$, and hence Proposition 2 of [Dan+24a] holds up to polylog(n) factors.

We can also compute the conditional expectation of the gradient $\boldsymbol{g}_t$:

Lemma D.4. *For any $t \geq 0$,*

$$\mathbb{E}_t \left[\boldsymbol{g}_t^\star \right] = \phi(m_t) \boldsymbol{w}_t^\star + \psi^{\text{corr}}(m_t) \boldsymbol{w}_t \tag{D.7}$$

where $\phi(m_t)$ and ψ^{corr} are two functions with Taylor expansion

$$\phi(m) = \sum_{k=0}^{\infty} c_{k+1} c_{k+1}^\star m^k \quad \textit{and} \quad \psi^{\text{corr}}(m) = \sum_{k=0}^{\infty} c_{k+2} c_k^\star m^k. \tag{D.8}$$

Further, we have

$$\mathbb{E}_t \left[\hat{\boldsymbol{g}}_t \right] = c^{\text{sq}} \boldsymbol{w}_t \quad \textit{with} \quad c^{\text{sq}} = \mathbb{E}_{z \sim \mathcal{N}(0,1)} \left[z \sigma(z) \sigma'(z) \right]. \tag{D.9}$$

Proof. The expectation of $\boldsymbol{g}_t^\star$ is a specialization of Lemma 4 from [Dan+24a] to $r = 1$. By the independence properties of Gaussians, $\mathbb{E}_t\left[\sigma(\langle \boldsymbol{w}_t, \boldsymbol{z}\rangle)\sigma'(\langle \boldsymbol{w}_t, \boldsymbol{z}\rangle)\langle \boldsymbol{z}, \boldsymbol{w}'\rangle\right] = 0$ as soon as $\boldsymbol{w}'$ is orthogonal to $\boldsymbol{w}_t$, hence the expectation of $\hat{\boldsymbol{g}}_t$ lies along $\boldsymbol{w}_t$, and the second result follows. □

In the following, we will denote $\psi^{\rm sq}(x) = \psi^{\rm corr}(x) - c^{\rm sq}$. A generic ψ will be used when no specialization is necessary, so that

$$\mathbb{E}_t\left[\boldsymbol{g}_t\right] = -\phi(m_t)\boldsymbol{w}^\star - \psi(m_t)\boldsymbol{w}_t. \tag{D.10}$$

D.1.3 A differential inequality for m_t

The structure of the proof is similar to the one of [BGJ21]. We define the following stopping times for $\zeta > 0$:

$$t_\zeta^+ = \min\{t \geq 0 : m_t \geq \zeta\}, \qquad t_\zeta^- = \min\{t \geq 0 : m_t \leq \zeta\}, \tag{D.11}$$

and the following γ-dependent time:

$$\tilde{t}_{\gamma,\zeta}^+ = \min\{t \geq 0 : \gamma m_t^{\ell-1} \geq \zeta\} \tag{D.12}$$

Our first goal is to show the following high-probability inequality:

Proposition D.1. *Define*

$$t_{\rm max} = \frac{n_b}{C_{\rm max} d \log(d)^{C_{\rm max}} \gamma^2}, \tag{D.13}$$

for some sufficiently large $C_{\rm max} > 0$. Then, for a sufficiently small choice of c_γ:

(i) If we are using the square loss, and $\gamma \leq c_\gamma(n_b d^{-\ell/2} \wedge 1)$, there exists $c, \eta > 0$ such that

$$\mathbb{P}\left(m_t \geq \frac{3}{4}m_0 + c\gamma\sum_{s=0}^{t-1} m_s^{\ell-1} \quad \forall t \leq t_\eta^+ \wedge t_{\rm max}\right) \geq 1 - ce^{-c\log(n)^2}. \tag{D.14}$$

(ii) If we are using the correlation loss, and $\gamma \leq c_\gamma n_b d^{-\ell/2}$, there exist $c, \varepsilon > 0$ such that

$$\mathbb{P}\left(m_t \geq \frac{3}{4}m_0 + c\gamma\sum_{s=0}^{t-1} m_s^{\ell-1} \quad \forall t \leq t_\eta^+ \wedge \tilde{t}_{\gamma,\varepsilon}^+ \wedge t_{\rm max}\right) \geq 1 - ce^{-c\log(n)^2}. \tag{D.15}$$

The rest of this section is devoted to show Proposition D.1. We define the following "good" event at time t:

$$\mathcal{E}_t \coloneqq \left\{|\langle \boldsymbol{g}_t, \boldsymbol{w}_t\rangle| \leq \frac{1}{2\gamma}\right\}$$

A (almost) deterministic update inequality We first expand the projection step to obtain a difference inequality for the process m_t. We write

$$\boldsymbol{g}_t = \langle \boldsymbol{g_t}, \boldsymbol{w_t} \rangle \boldsymbol{w}_t + \boldsymbol{g_t}^{\perp}, \tag{D.16}$$

where $\boldsymbol{g}_t^{\perp}$ is orthogonal to $\boldsymbol{w_t}$. Similarly to Equation D.10, we can compute the expectation of $\boldsymbol{g}_t^{\perp}$:

$$\mathbb{E}_t \left[\boldsymbol{g}_t \right] = -\phi(m_t)(\boldsymbol{w}_t^{\star} - m_t \boldsymbol{w}_t) \tag{D.17}$$

Lemma D.5. *For any $t \geq 0$, there exists a (random) constant c_t such that the following inequality holds:*

$$m_{t+1} \geq m_t - \gamma c_t \langle \boldsymbol{w}^{\star}, \boldsymbol{g}_t^{\perp} \rangle - \frac{\gamma^2 c_t^2 m_t \left\| \boldsymbol{g}_t^{\perp} \right\|^2}{2} - \frac{1}{2} \gamma^3 c_t^3 |\langle w^{\star}, \boldsymbol{g}^{\perp} \rangle| \left\| \boldsymbol{g}_t^{\perp} \right\|^2 . \tag{D.18}$$

Further, under the event $\mathcal{E}_t$, we have $1/2 \leq c_t \leq 2$.

Proof. We use the decomposition of eq. (D.16) and write

$$\boldsymbol{w}_t - \gamma \boldsymbol{g}_t = (1 - \gamma \langle \boldsymbol{g_t}, \boldsymbol{w_t} \rangle) \boldsymbol{w}_t - \gamma \boldsymbol{g}_t^{\perp}$$

As a result, if we define

$$c_t \coloneqq \frac{1}{1 - \gamma \langle \boldsymbol{g_t}, \boldsymbol{w_t} \rangle},$$

we have

$$\boldsymbol{w}_{t+1} = \frac{\boldsymbol{w}_t - \gamma c_t \boldsymbol{g}_t^{\perp}}{\left\| \boldsymbol{w}_t - \gamma c_t \boldsymbol{g}_t^{\perp} \right\|}$$

since the update equation (D.1) is invariant w.r.t scaling. Taking the scalar product of the above with $\boldsymbol{w}^{\star}$, we have

$$m_{t+1} = \frac{m_t - \gamma c_t \langle \boldsymbol{w}^{\star}, \boldsymbol{g}_t^{\perp} \rangle}{\left\| \boldsymbol{w}_t - \gamma c_t \boldsymbol{g}_t^{\perp} \right\|}. \tag{D.19}$$

Expanding the norm in the denominator, and using that $\|\boldsymbol{w}_t\| = 1$ and $\langle \boldsymbol{w}_t, \boldsymbol{g}_t^{\perp} \rangle = 0$:

$$\|\boldsymbol{w}_t - \gamma \boldsymbol{g}_t\| = \sqrt{1 + \gamma^2 c_t^2 \left\| \boldsymbol{g}_t^{\perp} \right\|^2}$$

By the convexity inequality $(1+x)^{-1/2} \geq 1 - x/2$, valid for all $x \geq 0$, Equation (D.19) becomes

$$m_{t+1} \geq \left(m_t - \gamma c_t \langle \boldsymbol{w}^{\star}, \boldsymbol{g}_t^{\perp} \rangle \right) \left(1 - \frac{\gamma^2 c_t^2}{2} \left\| \boldsymbol{g}_t^{\perp} \right\|^2 \right).$$

The lemma ensues upon expanding and rearranging the terms. □

The expansion in Lemma D.5 can be decomposed in two terms: the term linear in γ is a noisy *drift term*, that will drive the dynamics, and that we will decompose as a sum of a deterministic process and a martingale. All other terms in γ^2 or γ^3 are corrections that we bound with high probability.

The linear term We first control the term linear in γ. We can write

$$\langle \boldsymbol{w}^\star, \boldsymbol{g}_t^\perp \rangle = \langle \boldsymbol{w}^\star, \mathbb{E}_t \left[\boldsymbol{g}_t^\perp \right] \rangle + Z_t, \tag{D.20}$$

where $(Z_t)_{t\geq 0}$ is by definition a martingale difference sequence for the filtration $(\mathcal{F}_t)_{t\geq 0}$. The expectation term is straightforward to compute using (D.17):

$$\langle \boldsymbol{w}^\star, \mathbb{E}_t \left[\boldsymbol{g}_t^\perp \right] \rangle = -(1-m_t^2)\phi(m_t). \tag{D.21}$$

The contribution of the terms Z_t is bounded by the following lemma:

Lemma D.6. *There exists constants $c, C > 0$ such that with probability $1 - ce^{-c\log(n)^2}$,*

$$\sup_{1\leq t\leq T} \sum_{s=1}^{t} Z_s \leq \frac{C\log(d)^C\sqrt{T}}{\sqrt{n_b}} \tag{D.22}$$

Proof. The martingale increment Z_t is an average of n_b independent terms Z_t^ν, that satisfy $\|Z_t^\nu\|_{\psi_\alpha} \leq C$ for some $\alpha, C > 0$ by Assumption 4.1. As a result, if we define

$$B_\alpha = \sup_{t\geq 0} \|Z_t\|_{\psi_\alpha},$$

Theorem D.1 implies that

$$B_\alpha = \frac{\text{polylog}(d)}{\sqrt{n_b}}.$$

We now apply Theorem F.1 in [LJ21] with $z = \log(d)^{\frac{2(\alpha+2)}{\alpha}}\sqrt{T}B_\alpha$, which yields the exact bound needed. □

Bounding the corrections Our next step is to handle the higher-order corrections. We show the following lemma:

Lemma D.7. *Let $T \geq 0$, and $\eta < 1$. There exists a constant $C > 0$ such that for any $t \leq T$*

$$\mathbb{P}\left(\left\| \boldsymbol{g}_t^\perp \right\|^2 \leq C\left(\phi(m_t)^2(1-m_t^2)^2 + \frac{d\log(d)^C}{n_b} \right) \quad \forall t \leq t_\eta^+ \wedge T \right) \geq 1 - cTe^{-c\log(d)^2} \tag{D.23}$$

Proof. Fix some $t \in [T]$. We can write

$$\boldsymbol{g}_t^\perp = a(\boldsymbol{w}^\star - m\boldsymbol{w}) + \frac{1}{n}\sum_{\nu=1}^{n} f^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}^\nu \rangle)\sigma'(\langle \boldsymbol{w}_t, \boldsymbol{z}^\nu \rangle)\boldsymbol{z}^{\nu\perp}$$

where each $\boldsymbol{z}^{\nu\perp}$ is orthogonal to both $\boldsymbol{w}$ and $\boldsymbol{w}^\star$. From Lemmas 9 and 11 in [Dan+24a], with probability $1 - ce^{-c\log(d)^2}$,

$$\frac{1}{n}\sum_{\nu=1}^{n} f^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}^\nu\rangle)\sigma'(\langle \boldsymbol{w}_t, \boldsymbol{z}^\nu\rangle)\boldsymbol{z}^{\nu\perp} \leq C\frac{d\log(d)^C}{n_b}.$$

Now, we have

$$\langle \boldsymbol{g}_t^\perp, \boldsymbol{w}_t^\star\rangle^2 = a^2(1-m_t^2)^2 \quad \text{and} \quad \left\|a(\boldsymbol{w}^\star - m\boldsymbol{w})\right\|^2 = a^2(1-m_t^2),$$

hence

$$\left\|\boldsymbol{g}_t^\perp\right\|^2 \leq \frac{1}{1-m_t^2}\langle \boldsymbol{g}_t^\perp, \boldsymbol{w}_t^\star\rangle^2 + C\frac{d\log(d)^C}{n_b}.$$

It remains to notice that

$$\begin{aligned}\langle \boldsymbol{g}_t^\perp, \boldsymbol{w}_t^\star\rangle^2 &= (\langle \boldsymbol{w}^\star, \mathbb{E}_t\left[\boldsymbol{g}_t^\perp\right]\rangle + Z_t)^2 \\ &\leq 2(\langle \boldsymbol{w}^\star, \mathbb{E}_t\left[\boldsymbol{g}_t^\perp\right]\rangle^2 + Z_t^2) \\ &\leq \phi(m_t)^2(1-m_t^2)^2 + O\left(\frac{d}{n_b}\right).\end{aligned}$$

□

Putting it all together We now combine all the previous bounds into a unique proposition.

Proposition D.2. *Let $T \geq 0$. There exists constants $c, C > 0$ such that*

$$\begin{aligned}&\mathbb{P}\left(m_t \geq m_0 + \sum_{s=0}^{t-1}\Phi_{\text{drift}}(m_s) - C\Phi_{\text{noise}}(m_s) - CK(T) \quad \forall t \leq T \quad \middle| \quad \bigcap_{t\leq T}\mathcal{E}_t\right) \\ &\geq 1 - Te^{-c\log(n)^2}\end{aligned} \tag{D.24}$$

where Φ_{drift} and Φ_{noise} are given by

$$\Phi_{\text{drift}}(m) = \gamma(1-m^2)\phi(m), \tag{D.25}$$

$$\Phi_{\text{noise}}(m) = \gamma^2 m(1-m^2)\phi(m)^2 + \gamma^2 m\frac{d\log(d)^C}{n_b} + \gamma^3(1-m^2)^{3/2}\phi(m)^3, \tag{D.26}$$

$$K(T) = \frac{\gamma\log(d)^C\sqrt{T}}{\sqrt{n_b}} + \gamma^3 T\frac{d\log(d)^C}{n_b}. \tag{D.27}$$

Proof. By summing the inequality of Lemma D.5 for $0 \leq s \leq t-1$, we get

$$m_{t+1} \geq m_0 - \sum_{s=0}^{t-1}\gamma c_s\langle \boldsymbol{w}^\star, \boldsymbol{g}_s^\perp\rangle - \frac{\gamma^2 c_s^2 m_s\left\|\boldsymbol{g}_s^\perp\right\|^2}{2} - \frac{1}{2}\gamma^3 c_s^3\left\|\boldsymbol{g}_s^\perp\right\|^3. \tag{D.28}$$

The linear term is handled using the martingale decomposition (D.20) combined with the expectation computation of (D.21) and the bound of Lemma D.6 on the martingale contribution. The terms in γ^2 and γ^3 follow from Lemma D.7, as well as Lemma D.6 with $T = 1$, which implies that

$$|\langle \boldsymbol{w}^\star, \boldsymbol{g}_t^\perp \rangle \leq C \log(n)^C$$

for some $C > 0$. Finally, under the events $\mathcal{E}_s$ for $s \leq t$, we can replace every occurence of c_s by either $1/2$ or 2 depending on the sign of the corresponding term. □

Proof of Proposition D.1 The expressions of Lemma D.4 imply the following expansions near 0:

$$\phi(m) = c_\ell c_\ell^\star m^{\ell-1} + O(m^\ell) \quad \psi^{\text{corr}}(m) = O(m^\ell) \quad \psi^{\text{sq}}(m) = -c^{\text{sq}} + O(m^\ell) \tag{D.29}$$

As a result, there exists an $\eta > 0$ and constants $C, c > 0$ such that for any $m \leq \eta$,

$$cm^{\ell-1} \leq \phi(m) \leq Cm^{\ell-1} \qquad |\psi^{\text{corr}}(m)| \leq Cm^\ell \qquad |\psi^{\text{sq}}(m)| \leq C.$$

We first lower bound the drift inequality of Proposition D.2. Whenever $m_t \geq \eta$, we have

$$\Phi_{\text{drift}}(m_t) \geq p(\gamma m_t^{\ell-1}) - C\gamma^2 \frac{d}{n_b},$$

where $p(x) = x - C(x^2 + x^3)$. Define $\varepsilon > 0$ such that $p(x) > x/2$ on $[0, \varepsilon]$, so when $t \leq t_\eta^+ \wedge \tilde{t}_{\gamma,\varepsilon}^+$

$$\Phi_{\text{drift}}(m_t) \geq c\gamma m_t^{\ell-1} - C\gamma^2 \frac{d}{n_b} m.$$

When $\gamma \leq c_\gamma n_b d^{-\ell/2} \log(d)^{-C_\gamma}$,

$$\gamma \frac{d \log(d)^C}{n_b} m \leq c_\gamma (\sqrt{d})^{\ell-2} \leq \frac{cm^{\ell-1}}{2}$$

when $t \leq t_{\kappa/2\sqrt{d}}^-$, $C_\gamma \geq C$ and $c_\gamma \leq c/2(\kappa/2)^{\ell-2}$.

Having shown $\Phi_{\text{drift}}(m_t) \geq cm_t^{\ell-1}$, it remains to handle the constant terms in Proposition D.2. We can compute directly $K(t_{\text{max}})$, which yields

$$\begin{aligned} K(t_{\text{max}}) &= \frac{\log(d)^{C - C_{\text{max}}/2}}{C_{\text{max}}\sqrt{d}} + C\gamma \log(d)^{C - C_{\text{max}}} \frac{d}{n_b} \\ &\leq \frac{\log(d)^{C - C_{\text{max}}/2}}{C_{\text{max}}\sqrt{d}} + \frac{Cc_\gamma}{\log(d)^{C_{\text{max}}}} d^{-\frac{\ell}{2}} \\ &\leq \frac{m_0}{4}, \end{aligned}$$

by choosing c_γ small enough and C_{max} large enough.

Finally, we need to show that the events $\mathcal{E}_t$ occur with high probability. This is covered by the following lemma:

Lemma D.8. *Let $T \geq 0$. The following bounds hold:*

- *for the square loss, if $\gamma \leq c_\gamma$ for small enough c_γ,*

$$\mathbb{P}\left(\mathcal{E}_t \text{ holds for all } t \leq t_\eta^+ \wedge T\right) \geq 1 - Te^{-c\log(d)^2};$$

- *for the correlation loss, for small enough ε,*

$$\mathbb{P}\left(\mathcal{E}_t \text{ holds for all } t \leq t_\eta^+ \wedge \tilde{t}_{\gamma,\varepsilon}^+ \wedge T\right) \geq 1 - Te^{-c\log(d)^2}.$$

Proof. We begin with the case of the square loss. From the expression of the gradient expectation in (D.10), and the estimates (D.29),

$$\mathbb{E}_t\left[\langle \boldsymbol{w}_t, \boldsymbol{g}_t\rangle\right] = \psi^{\mathrm{sq}}(m_t) - \phi(m_t)m_t = O(1),$$

whenever $t \leq t_\eta^+$. Lemma D.1 applied to $\langle \boldsymbol{w}_t, \boldsymbol{g}_t\rangle$ implies that with probability $1 - ce^{-c\log(n)^2}$

$$|\langle \boldsymbol{w}_t, \boldsymbol{g}_t\rangle| \leq |\mathbb{E}_t\left[\langle \boldsymbol{w}_t, \boldsymbol{g}_t\rangle\right]| + \frac{C\log(d)^C}{\sqrt{n_b}} = O(1)$$

whenever $\mu > 0$. As a result, if $\gamma \leq c_\gamma$ for c_γ small enough, $\mathcal{E}_t$ holds. The proof for the correlation loss proceeds identically, noting this time that

$$\mathbb{E}_t\left[\langle \boldsymbol{w}_t, \boldsymbol{g}_t\rangle\right] = \psi^{\mathrm{corr}}(m_t) - \phi(m_t)m_t = O(1)$$

whenever $t \leq t_\eta^+ \wedge \tilde{t}_{\gamma,\varepsilon}^+$. □

D.1.4 From the linear regime to one-step recovery

We now prove Theorems 4.1 and 4.2. We focus on the case of the correlation loss; the square loss is identical apart from the additional $\gamma \leq c_\gamma$ requirement of Proposition D.1.

We first assume that $n_b = O(d^{\ell-1})$ and

$$\gamma \leq c_\gamma \log(d)^{-C_\gamma} \min\left(n_b d^{-\left(\frac{\ell}{2}\vee 1\right)}, \sqrt{\frac{n_b}{d}}\right) \tag{D.30}$$

for constants c_γ, C_γ to be chosen later. In particular, the condition $\gamma < c_\gamma n_b d^{-\ell/2}$ of Proposition D.1 is satisfied.

The linear regime The first part of the proof proceeds as in [BGJ21]. Define the function

$$t_\ell(d) = \begin{cases} 1 & \text{if } \ell = 1 \\ \log(d) & \text{if } \ell = 2 \\ d^{\frac{\ell}{2}-1} & \text{if } \ell > 2 \end{cases}, \tag{D.31}$$

and $t_{\mathrm{conv}} = \max(1, \gamma^{-1}t_\ell(d))$. Proposition D.1 as well as Section 5 from [BGJ21] implies the following lemma:

Lemma D.9. *There exists a constant $C > 0$ such that if $t_{\text{max}} \geq Ct_{\text{conv}}$, then with probability at least $1 - ce^{-c\log(n)^2}$,*

$$\tilde{t}_{\gamma,\varepsilon} \wedge t_\eta^+ \leq Ct_{\text{conv}}. \tag{D.32}$$

We therefore only need to check the condition $t_{\text{max}} \geq Ct_{\text{conv}}$. Plugging the expression for γ and t_{max}, we get

$$\frac{\gamma t_{\text{max}}}{t_\ell(d)} \geq \frac{n_b}{\gamma C_{\text{max}} \log(d)^{C_{\text{max}}+1} \gamma d^{1+(\ell/2-1)\vee 0}} \geq \frac{1}{c_\gamma C_{\text{max}}} \log(d)^{C_\gamma - C_{\text{max}} - 1}$$
$$t_{\text{max}} \geq \frac{n_b}{C_{\text{max}} \log(d)^{C_{\text{max}}} d\gamma^2} \geq \frac{1}{c_\gamma C_{\text{max}}} \log(d)^{2C_\gamma - C_{\text{max}} - 1}$$

Whenever $n_b = O(d^{\ell-1})$, by decreasing c_γ and increasing C_γ, for large enough d and any constant C we have

$$\frac{\gamma t_{\text{max}}}{t_\ell(d)} \leq C \quad \text{and} \quad t_{\text{max}} \geq C,$$

as requested in Lemma D.9.

Whenever $\gamma\eta^{\ell-1} \leq \varepsilon$, we have $\tilde{t}_{\gamma,\varepsilon} \geq t_\eta^+$ and hence the proof of Theorem 4.2 is complete. It thus remains to treat the converse case. Note that the latter only happens in the correlation loss case, since we can always choose c_γ such that $c_\gamma \eta^{\ell-1} \leq \varepsilon$; the dynamics of square loss SGD are therefore only in the linearized regime.

One-step recovery above $\tilde{t}_{\gamma,\varepsilon}$ We now treat the case where $\gamma\eta^{\ell-1} \geq \varepsilon$. For simplicity, let $t = \tilde{t}_{\gamma,\varepsilon}$, then Lemmas D.7 and D.6 imply that with probability $1 - ce^{-c\log(n)^2}$

$$\|\boldsymbol{g}_t\|^2 \leq C\left((|\phi(m_t)| + |\psi^{\text{corr}}(m_t)|)^2 + \frac{d}{n_b}\right) \leq C\left(m_t^{2\ell-2} + \frac{d}{n_b}\right)$$
$$\langle \boldsymbol{g}_t, \boldsymbol{w}^\star \rangle = \phi(m_t) + m_t\psi^{\text{corr}}(m_t) + O\left(\frac{\text{polylog}(d)}{\sqrt{n_b}}\right) \geq cm_t^{\ell-1} + O\left(\frac{\text{polylog}(d)}{\sqrt{n_b}}\right)$$

By definition of t, we have $1 \leq \varepsilon^{-1}\gamma m_t^{\ell-1}$, and whenever $\gamma \leq c_\gamma\sqrt{n_b/d}$ one has

$$\frac{\gamma}{\sqrt{n_b}} \leq c_\gamma d^{-1/2} \quad \text{and} \quad \gamma^2 \frac{d}{n_b} \leq c_\gamma^2.$$

Therefore, for large enough d,

$$m_t + \gamma\langle \boldsymbol{g}_t, \boldsymbol{w}^\star \rangle \geq \gamma\langle \boldsymbol{g}_t, \boldsymbol{w}^\star \rangle \geq c\gamma m_t^{\ell-1}$$
$$\|\boldsymbol{w}_t + \gamma \boldsymbol{g}_t\| \leq 1 + \gamma\|\boldsymbol{g}_t\| \leq 1 + C\gamma\left(m_t^{\ell-1} + \frac{d}{n_b}\right) \leq (C + \varepsilon^{-1} + c_\gamma^2\varepsilon^{-1})\gamma m_t^{\ell-1}$$

But by taking the scalar product of Equation (4.4) with $\boldsymbol{w}^\star$,

$$m_{t+1} = \frac{m_t + \gamma\langle \boldsymbol{g}_t, \boldsymbol{w}^\star \rangle}{\|\boldsymbol{w}_t + \gamma \boldsymbol{g}_t\|} \geq \frac{c}{C + \varepsilon^{-1} + c_\gamma^2\varepsilon^{-1}} =: \eta'.$$

Theorem 4.2 ensues by redefining $\eta = \min(\eta, \eta')$.

D.2 Informal derivation of Proposition 4.1

In this appendix we provide an informal derivation of the low-dimensional deterministic expressions describing the dynamics of the sufficient statistics (Proposition 4.1). While the formal rigorous characterization should in principle follow directly from [Vei+22], it requires a significant amount of work for full mathematical rigor.

Let $\mathcal{D}$ be the set of labeled data $\{\boldsymbol{z}^\nu, y^\nu\}_{\nu\in[n]}$, with label generated by:

$$y^\nu = f^\star(W^\star z^\nu) + \sqrt{\Delta}\xi^\nu,$$

where $W^\star \in \mathbb{R}^{k\times d}$ where we assume $k = O(1)$. We are implying that y^ν depends on $\boldsymbol{z}^\nu \sim \mathcal{N}(0, I_d)$ just through a low-dimensional representation (linear latent variable). $\xi^\nu \sim \mathcal{N}(0,1)$ is the artificial noise.

We can track the overlap matrix using standard manipulation. We introduce the local fields as:

$$\boldsymbol{\lambda}^\nu \coloneqq W\boldsymbol{z}^\nu \in \mathbb{R}^p, \quad \boldsymbol{\lambda}^{\star\nu} \coloneqq W^\star\boldsymbol{z}^\nu \in \mathbb{R}^k \qquad \forall\nu \in [n]$$

We fit these data using a two-layer neural network. Let the first layer weights be $W \in \mathbb{R}^{p\times d}$, the second layer weights $\boldsymbol{a} \in \mathbb{R}^p$; the full expression of the network is given by

$$f(\boldsymbol{z}) = \frac{1}{p}\sum_{j=1}^{p} a_j\sigma(\boldsymbol{w}_j^\top\boldsymbol{z}),$$

where w_j are the rows of W and σ is the activation function.

Since $\boldsymbol{z}^\nu$ is Gaussian and independent from $(W, W^\star)$, the pre-activations are jointly Gaussian vectors $(\boldsymbol{\lambda}^\nu, \boldsymbol{\lambda}^{\star\nu}) \sim \mathcal{N}(\mathbf{0}_{p+k}, \Omega)$ with covariance:

$$\Omega \coloneqq \begin{pmatrix} Q & M \\ M^\top & P \end{pmatrix} = \begin{pmatrix} WW^\top & WW^{\star\top} \\ W^\star W^\top & W^\star W^{\star T} \end{pmatrix} \in \mathbb{R}^{(p+k)\times(p+k)}$$

This is the low-dimensional matrix (its dimensions stay finite even when $d \to +\infty$) that contains all the information needed for the dynamics.

We are going to train the network with layer-wise SGD without replacement, using at each time step t a fresh new batch of size n_b:

$$\hat{\mathcal{R}}_t = \frac{1}{2n_b}\sum_{\nu=1}^{n_b}(y_t^\nu - f(\boldsymbol{z}_t^\nu))^2$$

A stated in the main text, we are interested in the *representation learning* phase. The gradient of the first layer weights

$$\nabla_{\boldsymbol{w}_j}\hat{\mathcal{R}}_t = -\frac{1}{pn_b}\sum_{\nu=1}^{n_b} a_j\sigma'(\lambda_{j,t}^\nu)\mathcal{E}_t^\nu\boldsymbol{z}_t^\nu \qquad \forall j \in [p]$$

where we defined for convenience the displacement vector

$$\mathcal{E}_t^\nu \coloneqq y_t^\nu - f(\boldsymbol{z}_t^\nu).$$

Initially, we focus on plain SGD, without normalizing the weights at every step. Let us take now one gradient step with learning rate γ:

$$\boldsymbol{w}_{j,t+1} = \boldsymbol{w}_{j,t} - \gamma \nabla_{\boldsymbol{w}_j} \hat{\mathcal{R}}_t \tag{D.33}$$

By combining the gradient update equation with the definitions of the matrices (W, W^*) we obtain the following dynamics:

$$\begin{aligned} M_{jr,t+1} - M_{jr,t} =& \frac{\gamma}{p n_b} a_j \sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^\nu) \lambda_r^\star \mathcal{E}_t^\nu \\ Q_{jl,t+1} - Q_{jl,t} =& \frac{\gamma}{p n_b} \sum_{\nu=1}^{n_b} \left(a_j \sigma'(\lambda_{j,t}^\nu) \lambda_{l,t}^\nu + a_l \sigma'(\lambda_{l,t}^\nu) \lambda_{j,t}^\nu \right) \mathcal{E}_t^\nu \\ &+ \frac{\gamma^2}{p^2 n_b^2} a_j a_l \sum_{\nu=1}^{n_b} \sum_{\nu'=1}^{n_b} \sigma'(\lambda_{j,t}^\nu) \sigma'(\lambda_{l,t}^{\nu'}) \mathcal{E}_t^\nu \mathcal{E}_t^{\nu'} \boldsymbol{z}_t^{\nu\top} \boldsymbol{z}_t^{\nu'} \end{aligned} \tag{D.34}$$

These equations introduce a discrete stochastic process $\{\Omega_t\}_{t\in\mathbb{N}}$ that describes the dynamics in a low-dimensional way. We also introduce the population loss as

$$\mathcal{R}_t = \frac{1}{2} \mathbb{E}_{\Omega_t} \left[\mathcal{E}^2 \right],$$

since it is the quantity telling us the performance of our trained network.

Handling the intra-batch correlations Up to now, we have followed the same derivation as the original Saad&Solla equations [SS95b], apart from the effective learning rate scaling. Using larger batches introduces some extra correlations terms that have to be taken into account. Let's split the second term of equation (D.34) for Q in 2:

$$\begin{aligned} \sum_{\nu=1}^{n_b} \sum_{\nu'=1}^{n_b} \sigma'(\lambda_{j,t}^\nu) \sigma'(\lambda_{l,t}^{\nu'}) \mathcal{E}_t^\nu \mathcal{E}_t^{\nu'} \boldsymbol{z}_t^{\nu\top} \boldsymbol{z}_t^{\nu'} =& \sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^\nu) \sigma'(\lambda_{l,t}^\nu) \mathcal{E}_t^{\nu 2} \boldsymbol{z}_t^{\nu\top} \boldsymbol{z}^\nu + \\ &+ \sum_{\nu=1}^{n_b} \sum_{\nu'=1, \nu'\neq\nu}^{n_b} \sigma'(\lambda_{j,t}^\nu) \sigma'(\lambda_{l,t}^{\nu'}) \mathcal{E}_t^\nu \mathcal{E}_t^{\nu'} \boldsymbol{z}_t^{\nu\top} \boldsymbol{z}_t^{\nu'} \end{aligned}$$

The first term is the standard gradient noise term that appears in Saad&Solla equations, while the second emerges from the large-batch, and has to be treated with new considerations. Let's introduce now the component of the student vectors orthogonal to the teacher space

$$W_t^\perp \coloneqq W_t - M_t P^{-1} W^\star \text{ and consequently } Q_t^\perp \coloneqq \left(W_t^\perp \right)^\top W_t^\perp = Q_t - M_t P^{-1} M_t^\top.$$

We can also define the local fields of this subspace, with the interesting property of being independent with the teacher ones

$$\boldsymbol{\lambda}^{\perp} \coloneqq W_t^{\perp} z \qquad \boldsymbol{\lambda}^{\perp} \sim \mathcal{N}(0, Q_t^{\perp}) \quad \text{Cov}[\boldsymbol{\lambda}^{\star}, \boldsymbol{\lambda}^{\perp}] = 0$$

It is possible to choose a set $\boldsymbol{v}_{\beta,t} \in \Big(\text{Span}\,(W_t^{\perp}) \cup \text{Span}\,(W^{\star})\Big)^{\perp}$ of orthonormal vectors, such that $\Big\{\boldsymbol{w}_r^{\star}, \boldsymbol{w}_{j,t}^{\perp}, \boldsymbol{v}_{\beta,t}\Big\}_{r\in[k],j\in[p],\beta\in[d-p-k]}$ is a basis of $\mathbb{R}^d$. Using the properties of the basis, we can write the identity matrix I_d as

$$I_d = (W^{\star})^{\top} P^{-1} W^{\star} + (W_t^{\perp})^{\top} (Q_t^{\perp})^{-1} W_t^{\perp} + \sum_{\beta=1}^{d-k-p} \boldsymbol{v}_{\beta,t} \boldsymbol{v}_{\beta,t}^{\top}$$

We insert the identity matrix $\boldsymbol{z}^{\top}\boldsymbol{z}$ with $\boldsymbol{z}^{\top} I_d \boldsymbol{z}$. By recalling that $\boldsymbol{\lambda}^{*} = W^{*}\boldsymbol{z}, \boldsymbol{\lambda}^{\perp} = W_t^{\perp}\boldsymbol{z}$, we arrive to:

$$\sum_{\nu=1}^{n_b} \sum_{\nu'=1,\nu'\neq\nu}^{n_b} \sigma'(\lambda_{j,t}^{\nu}) \sigma'(\lambda_{l,t}^{\nu'}) \mathcal{E}_t^{\nu} \mathcal{E}_t^{\nu'} \Bigg(\left(\boldsymbol{\lambda}^{\nu\star}\right)^{\top} P^{-1} \boldsymbol{\lambda}^{\nu'\star} + \\ + \left(\boldsymbol{\lambda}_t^{\perp\nu}\right)^{\top} (Q_t^{\perp})^{-1} \boldsymbol{\lambda}_t^{\perp\nu'} + \sum_{\beta=1}^{d-k-p} \langle \boldsymbol{v}_{\beta,t}, \boldsymbol{z}^{\nu} \rangle \langle \boldsymbol{v}_{\beta,t}, \boldsymbol{z}_t^{\nu'} \rangle \Bigg)$$

Now, exploiting the relation:

$$\boldsymbol{\lambda}_t^{\perp} = \boldsymbol{\lambda}_t - M_t P^{-1} \boldsymbol{\lambda}^{\star},$$

and noting that the indices ν and ν' are independent, all the sum we need to compute are just

$$\sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^{\nu}) \lambda_r^{\star} \mathcal{E}_t^{\nu} \qquad \sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^{\nu}) \lambda_{l,t}^{\nu} \quad \text{and} \quad \sum_{\nu=1}^{n_b} \sum_{\beta=1}^{d-k-p} \langle \boldsymbol{v}_{\beta,t}, \boldsymbol{z}_t^{\nu} \rangle$$

High dimensional limit In our analysis we consider the high-dimensional limit $d \to +\infty$ with the batch size going to infinite as well, with the scaling $n_b = n_0 d^{\mu}$. Note that when $\mu = 0$ the intra-batch correlation disappear and we fall back to standard Saad&Solla setting, given that the learning rate $\gamma = \gamma_0 d^{-\delta}$ is small enough. Indeed, we can informally say that all the sums above converge to their expected value

$$\frac{1}{n_b} \sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^{\nu}) \lambda_r^{\star} \mathcal{E}_t^{\nu} \to \mathbb{E}_{\Omega_t} \left[\sigma'(\lambda_j) \lambda_r^{\star} \mathcal{E}\right] = \psi_{jr,t} \tag{D.35}$$

$$\frac{1}{n_b} \sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^{\nu}) \lambda_{l,t}^{\nu} \to \mathbb{E}_{\Omega_t} \left[\sigma'(\lambda_j) \lambda_l \mathcal{E}\right] = \phi_{jl,t}^{\text{GF}} \tag{D.36}$$

$$\frac{1}{n_b} \sum_{\nu=1}^{n_b} \sigma'(\lambda_{j,t}^{\nu}) \sigma'(\lambda_{l,t}^{\nu}) \mathcal{E}_t^{\nu 2} \boldsymbol{z}_t^{\nu\top} \boldsymbol{z}^{\nu} \to d\mathbb{E}_{\Omega_t} \left[\sigma'(\lambda_j) \sigma'(\lambda_l) \mathcal{E}^2\right] = d\phi_{jl,t}^{\text{HD}} \tag{D.37}$$

$$\sum_{\nu=1}^{n_b} \sum_{\beta=1}^{d-k-p} \langle \boldsymbol{v}_{\beta,t}, \boldsymbol{z}_t^{\nu} \rangle \to 0$$

Moreover, using $\boldsymbol{\lambda}^\perp = \boldsymbol{\lambda} - MP^{-1}\boldsymbol{\lambda}^\star$ we have

$$\mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\mathcal{E}\boldsymbol{\lambda}^\perp\right] = \boldsymbol{\phi}^{\mathrm{GF}}_{j,t} - M_t P^{-1}\boldsymbol{\psi}_{j,t}$$

Plugging back in (D.34), we finally obtain

$$\begin{aligned}
M_{jr,t+1} - M_{jr,t} \approx& \frac{\gamma}{p} a_j \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda^\star_r \mathcal{E}\right] \\
Q_{jl,t+1} - Q_{jl,t} \approx& \frac{\gamma}{p}\mathbb{E}_{\Omega_t}\left[\left(a_j\sigma'(\lambda_j)\lambda_l + a_l\sigma'(\lambda_l)\lambda_j\right)\mathcal{E}\right] \\
&+ \frac{\gamma^2 d}{p^2 n_b} a_j a_l \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\sigma'(\lambda_l)\mathcal{E}^2\right] \\
&+ \mathbf{1}_{\{\mu\neq 0\}}\frac{\gamma^2}{p^2} a_j a_l \left(\mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\mathcal{E}\left(\lambda^\star\right)^\top\right] P^{-1}\mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_l)\mathcal{E}\lambda^\star\right]\right) \\
&+ \mathbf{1}_{\{\mu\neq 0\}}\frac{\gamma^2}{p^2} a_j a_l \left(\mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\mathcal{E}(\lambda^\perp)^\top\right]\left(Q_t^\perp\right)^{-1}\mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_l)\mathcal{E}\lambda^\perp\right]\right).
\end{aligned}$$

where the indicator function $\mathbf{1}_{\{\mu\neq 0\}}$ indicates that the last term is only present if the batch is large. If we want to make explicit all the dependencies in d ($\gamma = \gamma_0 d^{-\delta}, n_b = n_0 d^\mu$):

$$\begin{aligned}
M_{jr,t+1} - M_{jr,t} \approx& d^{-\delta}\frac{\gamma_0}{p} a_j \psi_{jr,t} = \Psi_{jr,t} \\
Q_{jl,t+1} - Q_{jl,t} \approx& d^{-\delta}\frac{\gamma_0}{p}\left(a_j\phi^{\mathrm{GF}}_{jl,t} + a_l^\tau \phi^{\mathrm{GF}}_{lj,t}\right) \\
&+ d^{-2\delta+1-\mu}\frac{\gamma_0^2}{p^2 n_0} a_j a_l \phi^{\mathrm{HD}}_{jl} \\
&+ d^{-2\delta}\frac{\gamma_0^2}{p^2} a_j a_l \boldsymbol{\psi}_j P^{-1}(\boldsymbol{\psi}_l)^\top \\
&+ d^{-2\delta}\frac{\gamma_0^2}{p^2} a_j a_l \left(\boldsymbol{\phi}^{\mathrm{GF}}_{j,t} - M_t P^{-1}\boldsymbol{\psi}_{j,t}\right)(Q_t - M_t P^{-1} M_t^\top)^{-1}\left(\boldsymbol{\phi}^{\mathrm{GF}}_{l} - M_t P^{-1}\boldsymbol{\psi}_{l}\right)^\top \\
&=: \Phi_{jl,t}
\end{aligned} \tag{D.38}$$

These equations are the starting point for all our considerations, both when investigating weakly correlation and when characterizing the exact dynamics.

Indeed, when we have a cold start we have to take into account that $\psi_{jr,t}, \phi^{\mathrm{GF}}_{jl,t}, \phi^{\mathrm{HD}}_{jl}$ can also go to 0 when $d \to +\infty$. A careful analysis for the leading terms of these equations around initializations will also give us information on the behaviour of the system, and ultimately it will allow to have rules on how to scale δ and μ to have the best performance. An example for these analysis for *generalized linear model* is provided in Appendix D.4.

On the other hand, when can also assume $\psi_{jr,t}, \phi^{\mathrm{GF}}_{jl,t}, \phi^{\mathrm{HD}}_{jl} = O_d(1)$ for all the dynamics[1] and use the equations for an asymptotic description. Clearly, depending on the values of δ and μ,

[1] This happens if $\ell \leq 1$ or when we provide an informed initialization.

not all terms are present in the limiting equations. A detailed discussion about this is provided in Section 4.4.

Spherical projection When we consider the spherical gradient descent, i.e., the modification of eq. (D.33)

$$\boldsymbol{w}_{j,t+1} = \frac{\boldsymbol{w}_{j,t} - \gamma \nabla_{\boldsymbol{w}_j} \hat{\mathcal{R}}_t}{||\boldsymbol{w}_{j,t} - \gamma \nabla_{\boldsymbol{w}_j} \hat{\mathcal{R}}_t||},$$

the overlap dynamics for the spherical large batch SGD can be then approximated as

$$\begin{aligned} M_{jr,t+1} - M_{jr,t} \approx& \Psi_{jr,t}(\Omega) - \frac{M_{jr,t}}{2}\Phi_{jj,t}(\Omega) \\ Q_{jl,t+1} - Q_{jl,t} \approx& \Phi_{jl,t}(\Omega) - \frac{1}{2}Q_{jl,t}\left(\Phi_{jj,t}(\Omega) + \Phi_{ll,t}(\Omega)\right) \end{aligned}$$

This derivation follows from a Taylor expansion of the denominator needed to project the update equations on the sphere. As final note, this approximation only holds when γ is vanishing when $d \to +\infty$: that's why we need $\gamma = o_d(1)$ in Proposition 4.1. When γ is too large, all the other orders of Taylor expansion play a role, and we can't have a simple expression for the exact evolution, even near initialization. Nevertheless, the first order expansion is a lower bound of the true dynamic that can provided guarantee of learning in some cases:

$$\begin{aligned} M_{jr,t+1} - M_{jr,t} \geq& \Psi_{jr,t}(\Omega) - \frac{M_{jr,t}}{2}\Phi_{jj,t}(\Omega) \\ Q_{jl,t+1} - Q_{jl,t} \geq& \Phi_{jl,t}(\Omega) - \frac{1}{2}Q_{jl,t}\left(\Phi_{jj,t}(\Omega) + \Phi_{ll,t}(\Omega)\right) \end{aligned} \tag{D.39}$$

D.3 Special case: committee machine with matching architecture

We consider a separable teacher, more precisely it is a committee machine with k hidden units, i.e.,

$$f_*(\boldsymbol{z}) = \frac{1}{k}\sum_{r=1}^{k} a_r^* \sigma(\lambda_r^\star)$$

where we additionally consider a matched architecture in which the student and teacher share the same activation function σ.

As we discussed in Section D.2, the activation and the target appear just in the expected values of

Equations (D.35),(D.36) and (D.37), that can be further simplified for matching architectures

$$\begin{aligned}
\psi_{jr,t} = \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda_r^\star \mathcal{E}\right] =& \frac{1}{k}\sum_{t'=1}^{k} a_{t'}^* \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda_r^*\sigma(\lambda_{t'}^*)\right] \\
&- \frac{1}{p}\sum_{s=1}^{p} a_s \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda_r^*\sigma(\lambda_s)\right] \\
\phi_{jl,t}^{\text{GF}} = \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda_l \mathcal{E}\right] =& \frac{1}{k}\sum_{t'=1}^{k} a_{t'}^* \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda_l\sigma(\lambda_{t'}^*)\right] \\
&- \frac{1}{p}\sum_{s=1}^{p} a_s \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\lambda_l\sigma(\lambda_s)\right] \\
\phi_{jl,t}^{\text{HD}} = \mathbb{E}_{\Omega_t}\left[\sigma'(\lambda_j)\sigma'(\lambda_l)\mathcal{E}^2\right] =& \frac{1}{k^2}\sum_{r,t'=1}^{k} a_r^* a_{t'}^* \mathbb{E}_{\Omega_t}\left[\sigma'\left(\lambda_j\right)\sigma'\left(\lambda_l\right)\sigma\left(\lambda_r^*\right)\sigma\left(\lambda_{t'}^*\right)\right] \\
&+ \frac{1}{p^2}\sum_{s,u=1}^{p} a_s a_u \mathbb{E}_{\Omega_t}\left[\sigma'\left(\lambda_j\right)\sigma'\left(\lambda_l\right)\sigma\left(\lambda_s\right)\sigma\left(\lambda_u\right)\right] \\
&- \frac{2}{pk}\sum_{s=1}^{p}\sum_{r=1}^{k} a_r^* a_s \mathbb{E}_{\Omega_t}\left[\sigma'\left(\lambda_j\right)\sigma'\left(\lambda_l\right)\sigma\left(\lambda_r^*\right)\sigma\left(\lambda_s\right)\right] \\
&+ \Delta\mathbb{E}_{\Omega_t}\left[\sigma'\left(\lambda_j\right)\sigma'\left(\lambda_l\right)\right]
\end{aligned}$$

In addition, we can also express the population risk as

$$\begin{aligned}
\mathcal{R}_t = \frac{1}{2}\mathbb{E}_{\Omega_t}\left[\mathcal{E}^2\right] =& \frac{\Delta}{2} + \frac{1}{p^2}\sum_{s,u}^{p} a_s a_u \mathbb{E}_{\Omega_t}\left[\sigma(\lambda_s)\sigma(\lambda_u)\right] + \\
&+ \frac{1}{k^2}\sum_{r,t}^{k} a_r^\star a_t^\star \mathbb{E}_{\Omega_t}\left[\sigma(\lambda_r^\star)\sigma(\lambda_t^\star)\right] - \\
&- \frac{2}{pk}\sum_{s,r=1}^{p,k} a_s a_r^\star \mathbb{E}_{\Omega_t}\left[\sigma(\lambda_s)\sigma(\lambda_r^\star)\right].
\end{aligned}$$

We introduce auxiliary functions to simplify the mathematical notations:

$$\begin{aligned}
I_2(\omega_{\alpha\alpha},\omega_{\alpha\beta},\omega_{\beta\beta}) &= \mathbb{E}\left[\sigma(\lambda_\alpha)\sigma(\lambda_\beta)\right] \\
I_3(\omega_{\alpha\alpha},\omega_{\alpha\beta},\omega_{\alpha\gamma},\omega_{\beta\beta},\omega_{\beta\gamma},\omega_{\gamma\gamma}) &= \mathbb{E}\left[\sigma'(\lambda_\alpha)\lambda_\beta\sigma(\lambda_\gamma)\right] \\
I_4(\omega_{\alpha\alpha},\omega_{\alpha\beta},\omega_{\alpha\gamma},\omega_{\alpha\delta},\omega_{\beta\beta},\omega_{\beta\gamma},\omega_{\beta\delta},\omega_{\gamma\gamma},\omega_{\gamma\delta},\omega_{\delta\delta}) &= \mathbb{E}\left[\sigma'(\lambda_\alpha)\sigma'(\lambda_\beta)\sigma(\lambda_\gamma)\sigma(\lambda_\delta)\right] \\
I_2^{\text{noise}}(\omega_{\alpha\alpha},\omega_{\alpha\beta},\omega_{\beta\beta}) &= \mathbb{E}\left[\sigma'(\lambda_\alpha)\sigma'(\lambda_\beta)\right].
\end{aligned}$$

where we introduced the correlation $\omega_{\alpha\beta} = \mathbb{E}[\lambda_\alpha\lambda_\beta]$, where (α, β) are indices running on either

the teacher or the student components. Dropping the time index for clarity, we finally obtain:

$$
\begin{aligned}
\psi_{jr} =& \frac{1}{k}\sum_{t=1}^{k} a_t^* I_3(Q_{jj}, M_{jr}, M_{jt}, P_{rr}, P_{rt}, P_{tt}) - \\
& - \frac{1}{p}\sum_{s=1}^{p} a_s I_3(Q_{jj}, M_{jr}, Q_{js}, P_{rr}, M_{sr}, Q_{ss}) \\
\phi_{jl}^{\mathrm{GF}} =& \frac{1}{k}\sum_{t=1}^{k} a_t^* I_3(Q_{jj}, Q_{jl}, M_{jt}, Q_{ll}, M_{lt}, P_{tt}) - \\
& - \frac{1}{p}\sum_{s=1}^{p} a_s I_3(Q_{jj}, Q_{jl}, Q_{js}, Q_{ll}, Q_{ls}, Q_{ss}) \\
\phi_{jl}^{\mathrm{HD}} =& \frac{1}{k^2}\sum_{r,t=1}^{k} a_r^* a_t^* I_4(Q_{jj}, Q_{jl}, M_{jr}, M_{jt}, Q_{ll}, M_{lr}, M_{lt}, P_{rr}, P_{rt}, P_{tt}) \\
& + \frac{1}{p^2}\sum_{s,u=1}^{p} a_s a_u I_4(Q_{jj}, Q_{jl}, Q_{js}, Q_{ju}, Q_{ll}, Q_{ls}, Q_{lu}, Q_{ss}, Q_{su}, Q_{uu}) \\
& - \frac{2}{pk}\sum_{s=1}^{p}\sum_{r=1}^{k} a_r^* a_s I_4(Q_{jj}, Q_{jl}, Q_{js}, M_{jr}, Q_{ll}, Q_{ls}, M_{lr}, Q_{ss}, M_{sr}, P_{rr}) \\
& + \Delta I_2^{\mathrm{noise}}(Q_{jj}, Q_{jl}, Q_{ll}) \\
\mathcal{R} =& \frac{\Delta}{2} + \frac{1}{p^2}\sum_{s,u}^{p} a_s a_u I_2(Q_{ss}, Q_{su}, Q_{uu}) + \frac{1}{k^2}\sum_{r,t}^{k} a_r^\star a_t^\star I_2(P_{rr}, P_{rt}, P_{tt}) \\
& - \frac{2}{pk}\sum_{s,r=1}^{p,k} a_s a_r^\star I_2(Q_{ss}, M_{sr}, P_{rr})
\end{aligned}
$$

When analyzing a matching architecture setting, we just need to specify I_2, I_3, I_4 and I_2^{noise}. In the following sections we provide the explicit expression for all the cases used in numerical simulation inside this paper.

D.3.1 Analytic case $\sigma = \mathrm{erf}\left(\cdot/\sqrt{2}\right)$

The expressions can be found in the appendix of [Vei+22].

D.3.2 Analytic case $\sigma = \mathbf{He}_2$

We report here the auxiliary functions:

$$I_2(\omega_{\alpha\alpha}, \omega_{\alpha\beta}, \omega_{\beta\beta}) = \mathbb{E}\left[(\lambda_\alpha^2 - 1)(\lambda_\beta^2 - 1)\right] = \omega_{\alpha\alpha}\omega_{\beta\beta} + 2\omega_{\alpha\beta}^2 - \omega_{\alpha\alpha} - \omega_{\beta\beta} + 1$$
$$I_3(\omega_{\alpha\alpha}, \omega_{\alpha\beta}, \omega_{\alpha\gamma}, \omega_{\beta\beta}, \omega_{\beta\gamma}, \omega_{\gamma\gamma}) = 2\mathbb{E}\left[\lambda_\alpha\lambda_\beta(\lambda_\gamma^2 - 1)\right] = 2\omega_{\alpha\beta}\omega_{\gamma\gamma} + 4\omega_{\alpha\gamma}\omega_{\beta\gamma} - 2\omega_{\alpha\beta}$$
$$I_4(\omega_{\alpha\alpha}, \omega_{\alpha\beta}, \omega_{\alpha\gamma}, \omega_{\alpha\delta}, \omega_{\beta\beta}, \omega_{\beta\gamma}, \omega_{\beta\delta}, \omega_{\gamma\gamma}, \omega_{\gamma\delta}, \omega_{\delta\delta}) = 4\mathbb{E}\left[\lambda_\alpha\lambda_\beta(\lambda_\gamma^2 - 1)(\lambda_\delta^2 - 1)\right]$$
$$I_2^{\text{noise}}(\omega_{\alpha\beta}) = 4\mathbb{E}\left[\lambda_\alpha\lambda_\beta\right] = 4\omega_{\alpha\beta}$$

We now work on the different terms:

$$\begin{aligned}4\mathbb{E}\left[\lambda^\alpha\lambda^\beta\left(\lambda^\gamma\right)^2\left(\lambda^\delta\right)^2\right] =& 4\omega_{\alpha\beta}\omega_{\gamma\gamma}\omega_{\delta\delta} + 8\omega_{\alpha\beta}\omega_{\gamma\delta}^2 + 8\omega_{\alpha\gamma}\omega_{\beta\gamma}\omega_{\delta\delta} + \\ & 16\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\gamma\delta} + 16\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\gamma\delta} + 8\omega_{\alpha\delta}\omega_{\beta\delta}\omega_{\gamma\gamma} \\ 4\mathbb{E}[\lambda_\alpha\lambda_\beta\lambda_\gamma^2] =& 4\omega_{\alpha\beta}\omega_{\gamma\gamma} + 8\omega_{\alpha\gamma}\omega_{\beta\gamma} \\ 4\mathbb{E}[\lambda_\alpha\lambda_\beta\lambda_\delta^2] =& 4\omega_{\alpha\beta}\omega_{\delta\delta} + 8\omega_{\alpha\delta}\omega_{\beta\delta} \\ 4\mathbb{E}[\lambda_\alpha\lambda_\beta] =& 4\omega_{\alpha\beta}\end{aligned}$$

And then we arrive to:

$$\begin{aligned}I_4 =& 4\omega_{\alpha\beta}\omega_{\gamma\gamma}\omega_{\delta\delta} + 8\omega_{\alpha\beta}\omega_{\gamma\delta}^2 + 8\omega_{\alpha\gamma}\omega_{\beta\gamma}\omega_{\delta\delta} \\ & 16\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\gamma\delta} + 16\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\gamma\delta} + 8\omega_{\alpha\delta}\omega_{\beta\delta}\omega_{\gamma\gamma} \\ & - 4\omega_{\alpha\beta}\omega_{\gamma\gamma} - 8\omega_{\alpha\gamma}\omega_{\beta\gamma} - 4\omega_{\alpha\beta}\omega_{\delta\delta} - 8\omega_{\alpha\delta}\omega_{\beta\delta} + 4\omega_{\alpha\beta}\end{aligned}$$

D.3.3 Analytic case $\sigma = \mathbf{He}_3$

We report the auxiliary function for this case below.

$$\begin{aligned}I_2^{\text{noise}}(\omega_{\alpha\alpha}, \omega_{\alpha\beta}, \omega_{\beta\beta}) \coloneqq& \mathbb{E}\left[(3\lambda_\alpha^2 - 3)(3\lambda_\beta^2 - 3)\right] \\ =& 9 - 9\omega_{\alpha\alpha} + 18\omega_{\alpha\beta}^2 - 9\omega_{\beta\beta} + 9\omega_{\alpha\alpha}\omega_{\beta\beta}\end{aligned}$$

$$\begin{aligned}I_2(\omega_{\alpha\alpha}, \omega_{\alpha\beta}, \omega_{\beta\beta}) \coloneqq& \mathbb{E}\left[(\lambda_\alpha^3 - 3\lambda_\alpha)(\lambda_\beta^3 - 3\lambda_\beta)\right] \\ =& 9\omega_{\alpha\beta} - 9\omega_{\alpha\alpha}\omega_{\alpha\beta} + 6\omega_{\alpha\beta}^3 - 9\omega_{\alpha\beta}\omega_{\beta\beta} + 9\omega_{\alpha\alpha}\omega_{\alpha\beta}\omega_{\beta\beta}\end{aligned}$$

$$\begin{aligned}I_3(\omega_{\alpha\alpha}, \omega_{\alpha\beta}, \omega_{\alpha\gamma}, \omega_{\beta\beta}, \omega_{\beta\gamma}, \omega_{\gamma\gamma}) \coloneqq& \mathbb{E}\left[(3\lambda_\alpha^2 - 3)\lambda_\beta(\lambda_\gamma^3 - 3\lambda_\gamma)\right] \\ =& -18\omega_{\alpha\beta}\omega_{\alpha\gamma} + 9\omega_{\beta\gamma} - 9\omega_{\alpha\alpha}\omega_{\beta\gamma} + 18\omega_{\alpha\gamma}^2\omega_{\beta\gamma} + \\ & 18\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\gamma\gamma} - 9\omega_{\beta\gamma}\omega_{\gamma\gamma} + 9\omega_{\alpha\alpha}\omega_{\beta\gamma}\omega_{\gamma\gamma}\end{aligned}$$

$$
\begin{aligned}
I_4(\cdots) \coloneqq & \mathbb{E}\left[(3\lambda_\alpha^2-3)(3\lambda_\beta^2-3)(\lambda_\gamma^3-3\lambda_\gamma)(\lambda_\delta^3-3\lambda_\delta)\right] \\
= & -162\omega_{\alpha\gamma}\omega_{\alpha\delta}+162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\beta}+324\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\gamma} \\
& -324\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\gamma}^2+324\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\delta}-162\omega_{\beta\gamma}\omega_{\beta\delta} \\
& +162\omega_{\alpha\alpha}\omega_{\beta\gamma}\omega_{\beta\delta}-324\omega_{\alpha\gamma}^2\omega_{\beta\gamma}\omega_{\beta\delta}-324\omega_{\alpha\delta}^2\omega_{\beta\gamma}\omega_{\beta\delta} \\
& -324\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\delta}^2+162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\gamma\gamma}-162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\beta}\omega_{\gamma\gamma} \\
& -324\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\gamma\gamma}-324\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma}+162\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma} \\
& -162\omega_{\alpha\alpha}\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma}+324\omega_{\alpha\delta}^2\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma}+324\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\delta}^2\omega_{\gamma\gamma} \\
& +81\omega_{\gamma\delta}-81\omega_{\alpha\alpha}\omega_{\gamma\delta}+162\omega_{\alpha\beta}^2\omega_{\gamma\delta} \\
& +162\omega_{\alpha\gamma}^2\omega_{\gamma\delta}+162\omega_{\alpha\delta}^2\omega_{\gamma\delta}-81\omega_{\beta\beta}\omega_{\gamma\delta} \\
& +81\omega_{\alpha\alpha}\omega_{\beta\beta}\omega_{\gamma\delta}-162\omega_{\alpha\gamma}^2\omega_{\beta\beta}\omega_{\gamma\delta}-162\omega_{\alpha\delta}^2\omega_{\beta\beta}\omega_{\gamma\delta} \\
& -648\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\gamma}\omega_{\gamma\delta}+162\omega_{\beta\gamma}^2\omega_{\gamma\delta}-162\omega_{\alpha\alpha}\omega_{\beta\gamma}^2\omega_{\gamma\delta} \\
& +324\omega_{\alpha\delta}^2\omega_{\beta\gamma}^2\omega_{\gamma\delta}-648\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\delta}\omega_{\gamma\delta}+1296\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\delta} \\
& +162\omega_{\beta\delta}^2\omega_{\gamma\delta}-162\omega_{\alpha\alpha}\omega_{\beta\delta}^2\omega_{\gamma\delta}+324\omega_{\alpha\gamma}^2\omega_{\beta\delta}^2\omega_{\gamma\delta} \\
& -81\omega_{\gamma\gamma}\omega_{\gamma\delta}+81\omega_{\alpha\alpha}\omega_{\gamma\gamma}\omega_{\gamma\delta}-162\omega_{\alpha\beta}^2\omega_{\gamma\gamma}\omega_{\gamma\delta} \\
& -162\omega_{\alpha\delta}^2\omega_{\gamma\gamma}\omega_{\gamma\delta}+81\omega_{\beta\beta}\omega_{\gamma\gamma}\omega_{\gamma\delta}-81\omega_{\alpha\alpha}\omega_{\beta\beta}\omega_{\gamma\gamma}\omega_{\gamma\delta} \\
& +162\omega_{\alpha\delta}^2\omega_{\beta\beta}\omega_{\gamma\gamma}\omega_{\gamma\delta}+648\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\delta}\omega_{\gamma\gamma}\omega_{\gamma\delta}-162\omega_{\beta\delta}^2\omega_{\gamma\gamma}\omega_{\gamma\delta} \\
& +162\omega_{\alpha\alpha}\omega_{\beta\delta}^2\omega_{\gamma\gamma}\omega_{\gamma\delta}-324\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\gamma\delta}^2+324\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\beta}\omega_{\gamma\delta}^2 \\
& +648\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\gamma\delta}^2+648\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\gamma\delta}^2-324\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\delta}^2 \\
& +324\omega_{\alpha\alpha}\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\delta}^2+54\omega_{\gamma\delta}^3-54\omega_{\alpha\alpha}\omega_{\gamma\delta}^3 \\
& +108\omega_{\alpha\beta}^2\omega_{\gamma\delta}^3-54\omega_{\beta\beta}\omega_{\gamma\delta}^3+54\omega_{\alpha\alpha}\omega_{\beta\beta}\omega_{\gamma\delta}^3 \\
& +162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\delta\delta}-162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\beta}\omega_{\delta\delta}-324\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\delta\delta} \\
& +324\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\gamma}^2\omega_{\delta\delta}-324\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\delta\delta}+162\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\delta\delta} \\
& -162\omega_{\alpha\alpha}\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\delta\delta}+324\omega_{\alpha\gamma}^2\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\delta\delta}-162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\gamma\gamma}\omega_{\delta\delta} \\
& +162\omega_{\alpha\gamma}\omega_{\alpha\delta}\omega_{\beta\beta}\omega_{\gamma\gamma}\omega_{\delta\delta}+324\omega_{\alpha\beta}\omega_{\alpha\delta}\omega_{\beta\gamma}\omega_{\gamma\gamma}\omega_{\delta\delta}+324\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma}\omega_{\delta\delta} \\
& -162\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma}\omega_{\delta\delta}+162\omega_{\alpha\alpha}\omega_{\beta\gamma}\omega_{\beta\delta}\omega_{\gamma\gamma}\omega_{\delta\delta}-81\omega_{\gamma\delta}\omega_{\delta\delta} \\
& +81\omega_{\alpha\alpha}\omega_{\gamma\delta}\omega_{\delta\delta}-162\omega_{\alpha\beta}^2\omega_{\gamma\delta}\omega_{\delta\delta}-162\omega_{\alpha\gamma}^2\omega_{\gamma\delta}\omega_{\delta\delta} \\
& +81\omega_{\beta\beta}\omega_{\gamma\delta}\omega_{\delta\delta}-81\omega_{\alpha\alpha}\omega_{\beta\beta}\omega_{\gamma\delta}\omega_{\delta\delta}+162\omega_{\alpha\gamma}^2\omega_{\beta\beta}\omega_{\gamma\delta}\omega_{\delta\delta} \\
& +648\omega_{\alpha\beta}\omega_{\alpha\gamma}\omega_{\beta\gamma}\omega_{\gamma\delta}\omega_{\delta\delta}-162\omega_{\beta\gamma}^2\omega_{\gamma\delta}\omega_{\delta\delta}+162\omega_{\alpha\alpha}\omega_{\beta\gamma}^2\omega_{\gamma\delta}\omega_{\delta\delta} \\
& +81\omega_{\gamma\gamma}\omega_{\gamma\delta}\omega_{\delta\delta}-81\omega_{\alpha\alpha}\omega_{\gamma\gamma}\omega_{\gamma\delta}\omega_{\delta\delta}+162\omega_{\alpha\beta}^2\omega_{\gamma\gamma}\omega_{\gamma\delta}\omega_{\delta\delta} \\
& -81\omega_{\beta\beta}\omega_{\gamma\gamma}\omega_{\gamma\delta}\omega_{\delta\delta}+81\omega_{\alpha\alpha}\omega_{\beta\beta}\omega_{\gamma\gamma}\omega_{\gamma\delta}\omega_{\delta\delta}
\end{aligned}
$$

D.4 Weak recovery with Generalized Linear Models

In this section, we restrict our analysis to matching architectures with $p = k = 1$, i.e. *Generalized Linear Models* (GLMs). Moreover, we consider as activation function the Hermite polynomials $\sigma = \mathrm{He}_\ell$, so that we can have control on the information exponent of the problem. Finally, the training algorithm is *projected SGD*, given by Equation (4.5). We will also assume that $a = a_\star = 1$ throughout all the dynamics.

Let us start by noticing that the set of sufficient statistics reduces to just one single parameter $m = \langle \boldsymbol{w}, \boldsymbol{w}_\star \rangle$. Retracing backward all the steps of the Sections D.3 and D.2 up to (D.39), we can obtain the lower bound for the update of m. As examples the explicit equation for $\sigma = \mathrm{He}_2$ is

$$\begin{aligned} m_{t+1} - m_t \geq & \gamma_0 d^{-\delta} \Bigg[4m_t - 4m_t^3 - d^{-\delta}\gamma_0 \mathbf{1}_{\{\mu \neq 0\}} \left(8m_t - 8m_t^3\right) + \\ & + d^{-\delta+1-\mu} \frac{\gamma_0}{n_0} \left(24m_t - 24m_t^3 + 2m_t^2 \Delta\right) \Bigg], \end{aligned}$$

while for $\sigma = \mathrm{He}_3$ is

$$\begin{aligned} m_{t+1} - m_t \geq & \gamma_0 d^{-\delta} \Bigg[18m_t^2 - 18m_t^4 - d^{-\delta}\gamma_0 \mathbf{1}_{\{\mu \neq 0\}} \left(162m_t + 324m_t^4 - 162m_t^5\right) + \\ & + d^{-\delta+1-\mu} \frac{\gamma_0}{n_0} \left(-1728m_t - 648m_t^3 + 3348m_t^4 - 972m_t^5 - 9\Delta m_t^3\right) \Bigg]. \end{aligned}$$

In general, for an Hermite polynomial activation He_ℓ, the equation of the evolution of m around $m = 0$ is given by

$$m_{t+1} - m_t \geq d^{-\delta} \beta_\ell m^{\ell-1} - d^{-\delta} \left(d^{-\delta+1-\mu}\alpha_\ell + d^{-\delta}\mathbf{1}_{\{\mu \neq 0\}}\phi_\ell\right) m \tag{D.40}$$

where we fixed $\gamma_0 = n_0 = 1$ for simplicity; $\alpha_\ell, \beta_\ell, \phi_\ell$ are constants. For computing the full equations for any generic ℓ, with the constants values, we refer to the Mathematica notebook published in the repository of this work.

At initialization, $m_0 = 1/\sqrt{d}$. The crucial observation is that a sufficient condition to escape initialization is to have equation (D.40) being *expansive*, namely $\Delta m > 0$ for m close to zero. This can be true if and only if $\left(d^{-\delta+1-\mu}\alpha_\ell + d^{-\delta}\mathbf{1}_{\{\mu \neq 0\}}\phi_\ell\right) m$ is negligible when compared to $\beta_\ell m^{\ell-1}$, so that the equation for m is lower-bounded by

$$\frac{\Delta m}{\Delta t} \geq \beta_\ell m^{\ell-1} + \text{h.o.t} \quad \text{with} \quad \Delta t = d^{-\delta} \tag{D.41}$$

Assuming that $\gamma = o_d(1)$, the bound becomes tight and we can also derive some sharp characterization of the escaping time. By simple arguments on differential equations, we can claim that

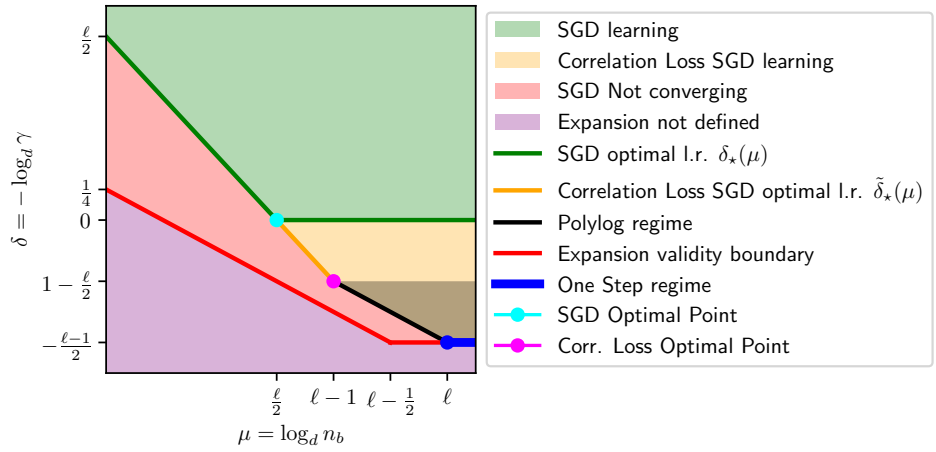


Figure D.1: **Phase diagram for the learning rate:** The plot identifies different learning behaviors of standard SGD and *Correlation Loss SGD* for different values of learning rate and batch size when considering randomly initialized networks, i.e. $m_0 = O(1/\sqrt{d})$.

the order of magnitude of steps needed to escape the initial mediocrity is given by

$$T = \begin{cases} O_d\left(\frac{1}{\Delta t}\right) & \ell = 1 \\ O_d\left(\frac{\log(1/m_0)}{\Delta t}\right) & \ell = 2 \\ O_d\left(\frac{1}{m_0^{\ell-2}\Delta t}\right) & \ell \geq 3 \end{cases},$$

remembering that $m_0 = 1/\sqrt{d}$ and $\Delta t = d^{-\delta}$, these lead to

$$\log_d T \sim \begin{cases} \max(\delta, 0) & \ell = 1 \\ \log\log d + \delta & \ell = 2 \\ \delta - 1 + \ell/2 & \ell \geq 3 \end{cases}. \tag{D.42}$$

It's clear that to escape as fast as possible, we want δ to be the smallest possible, or in other words, having the learning rate as large as possible. Obviously, δ is constrained by the values that make equation (D.41) true (or equivalently by the assumptions of the formal proof in Appendix D.1). The phase diagram of the allowed value of δ and μ is summarized in Figure D.1: the green region is where the equations for m is expansive, the red and the yellow region is where the equations in attractive, so there is no escaping, the purple region is where we can't do expansion because the learning rate is too large and the process diverge. Figure 4.1 in the main text shows the same result in terms of T and n_b, when $\ell \geq 3$.

D.4.1 Correlations loss SGD

When using *Correlation Loss SGD*, Equation (D.40) rewrite as

$$\frac{\Delta m}{\Delta t} \geq \beta_\ell m^{\ell-1} - d^{-\delta+1-\mu}\alpha_\ell m$$

effectively removing a constraint on the possible values of δ. The modified version of SGD can use smaller values of δ for escaping the initial condition, reaching regions in the phase diagram that are not allowed for SGD: this is colored in yellow in Figures D.1 and 4.1. Of course, if the learning rate becomes too large, all the theory does not work anymore (purple region in the diagram). In Figure 4.1 we show that the number of steps needed to weakly recover can be pushed down to be smaller than any power scaling with d (black and blue line on the x-axis). The picture becomes clearer if we look at the same diagram in terms of (μ, δ): Correlation Loss SGD can be used with learning very large learning rates ($1 - \ell/2 > \delta > -(\ell-1)/2$) such that the escaping times is $T = O(\text{polylog}(d))$, as proved in Theorem 4.2. We believe that the true number of steps is actually $T = O(\log(d))$, but we could not find any formal proof; in Section D.5.4, we were able to show that for $\ell = 2$ we have $T = O(\log(d))$, relying the result on numerical integration of our asymptotic theory. Lastly, if the learning rate is of order $\gamma = O(d^{-\delta}) = O(d^{\ell - 1/2})$ we recover the result of [Dan+24a]: the target can be weakly recovered in just one step, when the batch size is $n_b > O(d^\ell)$.

D.4.2 Simple example: retrieving [BGJ21]

In this section, we want to show how to find the same result presented in [BGJ21] starting from our formalism. There, online one-pass SGD is considered, meaning $n_b = 1 \implies \mu = 0$ in our context. Moreover, a vanishing learning rate is assumed, which implies $\delta > 0$, and all the bounds for the evolutions of m are tight. The condition for expansiveness of equation (D.40) becomes

$$m_0^{\ell-1} >> d^{-\delta+1} m_0 \implies (\ell-1)\log_d m_0 \geq -\delta+1+\log_d m_0 \implies \delta \geq 1+(2-\ell)\log_d m_0$$

Plugging in $m_0 = 1/\sqrt{d}$ we finally get $\delta \geq \ell/2$, where the equality is the best possible value of the learning rate in order to make the escaping faster. Note that for $\ell = 1$ we are also bounded from Lemma D.9, so $\delta \geq 1$. Combining with (D.42), finally gives us the minimal number of steps needed

$$T \sim \begin{cases} d & \ell = 1 \\ d\log d & \ell = 2 \\ d^{\ell-1} & \ell \geq 3 \end{cases}. \tag{D.43}$$

This result matches [BGJ21].

D.4.3 Extension to $p > 1$

The extension to general two-layer network student functions ($p > 1$), while keeping always the target fixed to be a single-index one, can be readily done by performing an analysis similar to the above. The considerations on the weak recovery trade-offs done in the previous sections are not changed upon re-scaling the learning rate with the hidden layer size $p = O(1)$, i.e. $\gamma_{\text{2LNN}}/p =$

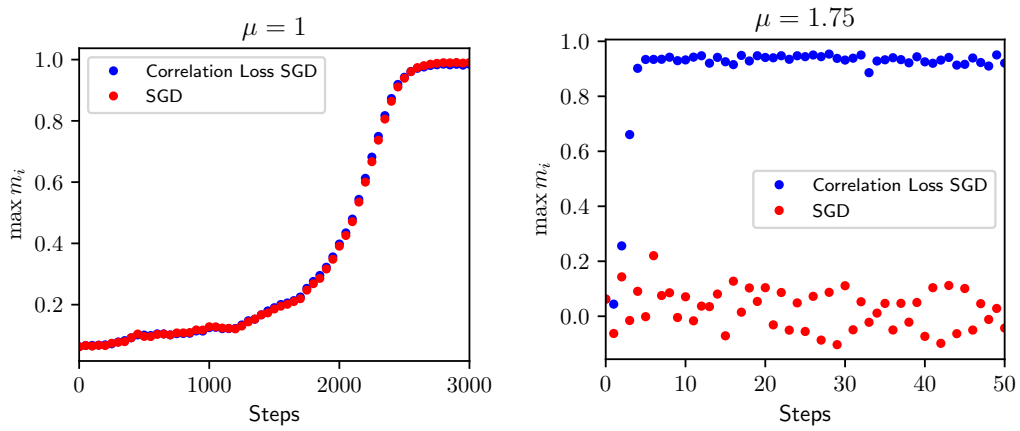


Figure D.2: learning single-index teacher with a wide student, when information exponent is $\ell = 3$: $f^\star(\boldsymbol{x}) = \mathrm{He}_3(\boldsymbol{w}^\star \cdot \boldsymbol{x})$, $f(\boldsymbol{x}) = 1/4 \sum_{i=1}^4 \mathrm{He}_3(\boldsymbol{w}_i \cdot \boldsymbol{x})$. Our theory extends to this case, showing that when $\mu > \ell/2$ only correlation loss can weakly recover the target. ($d = 256, \gamma = \gamma_0 \cdot p n_b d^{-\ell/2}$)

γ_{GLM}. Therefore, the scaling laws detailed in the phase diagram (Figure 4.1) are not modified, and only prefactors, i.e. quantity not scaling with the input dimension, change with respect to the $p = 1$ case. We illustrate this phenomenon numerically in Figure D.2. We leave the detailed theoretical analysis of the $p > 1$ case for future work, with particular attention to the limit $p \to \infty$ which we believe is an interesting avenue of future research.

D.5 Additional numerical investigation

In this appendix, we provide additional details on the numerical implementations presented in the main text, along with further explorations. The code to reproduce representative figures is available in our GitHub repository[2].

D.5.1 Cold start for multi-index models

The theoretical considerations for weak recovery under cold starts presented in Theorems 4.1&4.2 are proven rigorously just for one-hidden neuron network learning single-index targets ($p = k = 1$); this section aims to provide arguments to generalize this to the multi-index case.

Note that for single-index models the initial saddle is the only critical point where the algorithm can get stuck during the dynamics, while this is not true in general for multi-index settings. Indeed, after having weakly recovered a subspace of the span of the target weights, the learning dynamics can encounter another saddle of the loss function; this behavior is known as saddle-to-saddle dynamics [Jac+21]. In this manuscript, we focus on escaping from the saddle at initialization, leaving further explorations of the dynamics to future work. We follow [AAM23; Dan+24a]

[2] `https://github.com/IdePHICS/batch-size-time-complexity-tradeoffs`

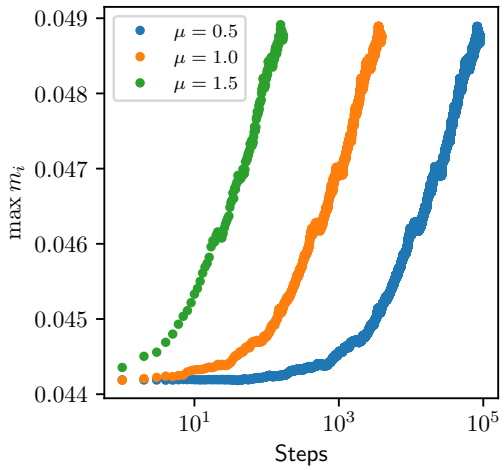


Figure D.3: **Multi-index large-batch benefits:** Comparison between the performance of plain SGD learning multi-index model, for different values of μ. The target is $h^\star(z_1, z_2, z_3) = \tanh(z_1 z_2 z_3)$, while the student is a wide 2-layer network $f(\boldsymbol{x}) = \frac{1}{p}\sum_{i=1}^{p} \tanh(\boldsymbol{w}_i \cdot \boldsymbol{x})$ (hence $\ell = 3, p = 30, k = 3, d = 512$). Using a larger batch speeds up the weak-correlation time even when the target is multi-index, and it is learned with a non-matching architecture.

where the authors generalize the concept of Information Exponent (defined in Definition 4.1) to the multi-index setting (See Definition 1 of [AAM23] and Definition 3 of [Dan+24a]), let us call this quantity the *Leap Index* of the target. We expect that, as long as the dynamics around the saddle at initialization is analyzed, one can substitute the Information Exponent (ℓ) of the teacher in the single-index phase diagram in Figure 4.1 with the Leap Index of the target. We explore the Time / Complexity tradeoffs in Figure D.3 for a fixed teacher function with Leap Index equal to 3: we observe a relevant decrease in the iterations needed to weakly recover the target subspace as the batch size is increased.

D.5.2 Behavior of *Spherical SGD*

In many theoretical work [BGJ21; AAM23], the algorithm used during training uses the *spherical gradient* instead of the simple one. The update rule used instead of Equation (D.33) is

$$\boldsymbol{w}_{j,t+1} = \frac{\boldsymbol{w}_{j,t} - \gamma\left(I_d - \boldsymbol{w}_{j,t}\boldsymbol{w}_{j,t}^\top\right)\nabla_{\boldsymbol{w}_{j,t}}\hat{\mathcal{R}}_t}{\left\|\boldsymbol{w}_{j,t} - \gamma\left(I_d - \boldsymbol{w}_{j,t}\boldsymbol{w}_{j,t}^\top\right)\nabla_{\boldsymbol{w}_{j,t}}\hat{\mathcal{R}}_t\right\|} \qquad \forall t \in [T], \forall j \in [p] \tag{D.44}$$

In practice, only the gradient component orthogonal to the weights is taken into account. This algorithm is particularly convenient for theoretical analysis because it is easier to find a lower bound for the evolution of m_t, since it is always true that

$$\left\|\boldsymbol{w}_{j,t} - \gamma\left(I_d - \boldsymbol{w}_{j,t}\boldsymbol{w}_{j,t}^\top\right)\nabla_{\boldsymbol{w}_{j,t}}\hat{\mathcal{R}}_t\right\| \geq 1,$$

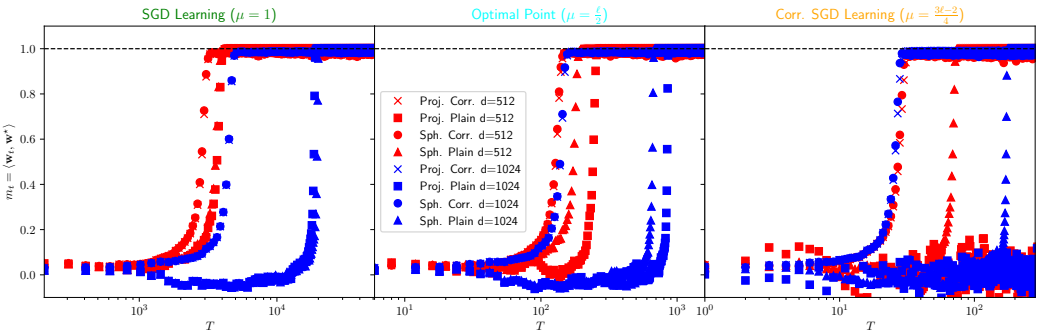


Figure D.4: **Correlation Loss SGD weak recovery:** Comparison between the performance of plain SGD, the Correlation Loss SGD and Spherical SGD, in different regions of the phase diagram, and for different sizes d. The plot shows the test error as a function of the optimization steps. Both the teacher and the student activation functions are fixed to $\sigma = h^\star = \mathrm{He}_3$, so the information exponent is $\ell = 3$. In all the three plots we vary the value of μ, while $\delta = \ell/2 - \mu$. Spherical SGD learns even in regions forbidden for plain SGD, as the Correlation Loss does. Note that the Spherical Correlation Loss is equivalent to the Projected Correlation Loss in all the regimes.

while its analogous for Projected SGD does not hold.

In this section, we want to show that *Spherical SGD* is behaving like *Correlation Loss SGD* when γ is not vanishing, namely that is possible to escape mediocrity when the batch size is sufficiently large. For small batch size, Projected SGD and Spherical SGD coincide, while when $\delta < 0$ their behaviors are drastically different, and only the latter is able to escape mediocrity taking advantage of the large learning rate; a gap between the two is already noticeable at the *Optimal Point*, where they both escape but the spherical is slightly faster. Finally, note that it is possible to introduce a *Correlation Loss Spherical SGD*, by changing the loss in the same way as the usual *Correlation Loss SGD*. There is no practical difference between the two algorithms when working with correlation loss.

D.5.3 *Adaptive SGD*: combining Correlation Loss SGD with plain SGD

Despite these benefits, the correlation loss is not a good choice to fully learn the target. In this subsection, we explore the idea of combining the two algorithms to escape fast with correlation loss, and then reach the global minimum with the MSE loss. We will call the combination of these two algorithms *Adaptive SGD*.

We are going to test in the simplest case possible: GLM with He_3 as activation function (we remark that there is no benefit in using Correlation Loss SGD over plain SGD when $\ell \leq 2$). If we run the algorithm for multi-index models, it would help to escape the initial saddle, but the algorithm may get stuck in another critical point that is not the global minimum. The study on how to escape fast from a critical point other than the initial one goes beyond the scope of this paper. Our *Adaptive SGD* procedure works as follows:

(i) Make a Correlation Loss SGD step;

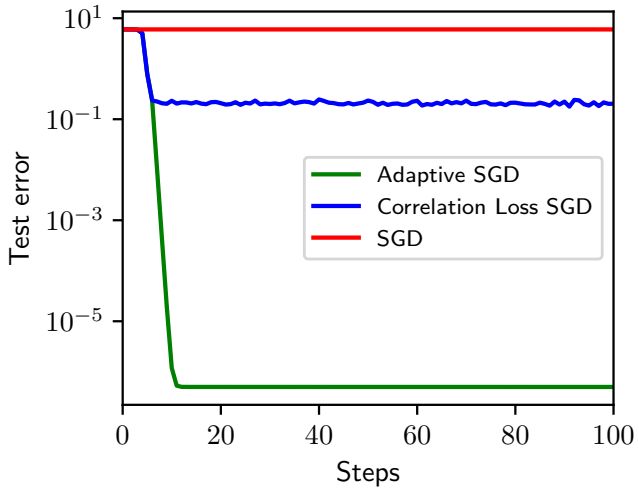


Figure D.5: **Adaptive SGD:** The plot compares the performance of SGD and Correlation loss SGD, algorithms with Adaptive SGD; this protocol consists of first using correlation loss SGD to achieve weak recovery, and then switch to adaptive SGD for learning the target. ($\ell = 3, \mu = 1.85, \delta = \mu - \frac{\ell}{2}, \Delta = 10^{-6}$).

(ii) If the Loss is smaller than 60% of the initial loss, jump to Step 3, otherwise go back to Step 1;

(iii) Reduce the learning rate of a factor 0.995 and do a Standard SGD step;

(iv) If converged stop, otherwise go back to Step 3.

The learning rate is progressively reduced because the plain SGD requires a lower learning rate compared to the one used by correlation loss to escape fast. Certainly, one can design a much more powerful algorithm than the one we present, but the goal here is just to show that the combination of the two is beneficial, and not to find the best possible one. Figure D.5 shows how the Adaptive SGD Algorithm is the best one when fully learning the target.

D.5.4 Polylog regime example

We showed in Section 4.3 that it is possible to push down the number of steps needed to weakly recover the target until it is growing less than any power law. In order to achieve this, we need to run *Correlation Loss SGD* with $n_b > d^{\ell-1}$ and $\gamma > O(d^{1-\ell/2})$, as pictured by Figure D.1. Proposition 4.1 shows that our theory for cold start, based on expansion of the process (D.38), is not valid, among other conditions, when $\delta < 0$. Therefore, the only case where we can simultaneously observe the *polylog regime* and have an exact asymptotic description for the full dynamics is when $\ell = 2, \gamma = O(1)$ and $n_b = O(d)$. Let's stick for simplicity with a GLM whose activation is He_2. Note that the total sample complexity is always $N = n_b T = O(d \log d)$.

Figure D.6 shows a numerical test of our theory in this particular case. We see that as d grows, also the time needed to escape initial conditions grows. In the right part of the Figure we show that the exact dependence is $T = O(\log d)$, that is indeed a polylog law.

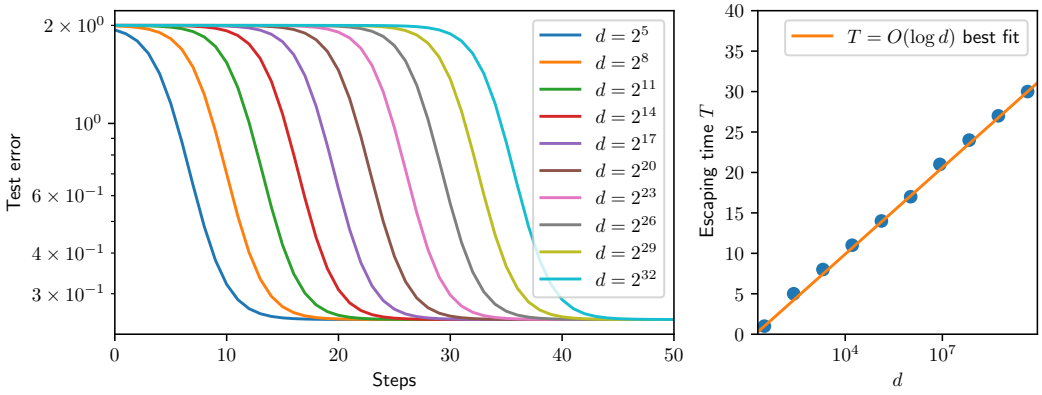


Figure D.6: **Phase retrieval with large batch size:** Numerical integration of the process (D.38), for $f = f^\star = \mathrm{He}_2, \gamma = O(1)$ and $n_b = O(d)$. The escaping time dependence on the number of time steps is a $T = O(\log d)$: we claim this to be valid for all the *polylog regime*.

D.5.5 Large-batch corrections to asymptotic dynamics

Although disappearing when taking the limit, the terms of evolution process (D.38) coming from intra-batch correlation are useful for providing a better description at large but finite d. Effectively, they are behaving as a first correction to the asymptotic limit.

In this section, we aim to provide numerical arguments about the importance of intra-batch correlations at finite d. We stick with the GLM setting, with erf as activation function. Note that since the information exponent of this target is 1, there is no mediocrity at initialization, we can set $m = \boldsymbol{w}^\top \boldsymbol{w}^\star = 0$ without falling in the *cold start regime*. Figure D.7 shows simulations for different values of d (dots), accompanied by the full process dynamic that includes the intra-batch correlation terms (dashed lines); the asymptotic solution of the differential equations (4.20) is the continuous black line. To enlighten the process even more, we also shows the difference between the asymptotic solution, at the actual finite d one on the right part of the figure. We see that the full process solution always match with the actual project SGD simulation; most importantly, when d grows the simulations are getting closer and closer to the asymptotic solution, confirming that the large batch plays no effect in high-dimension.

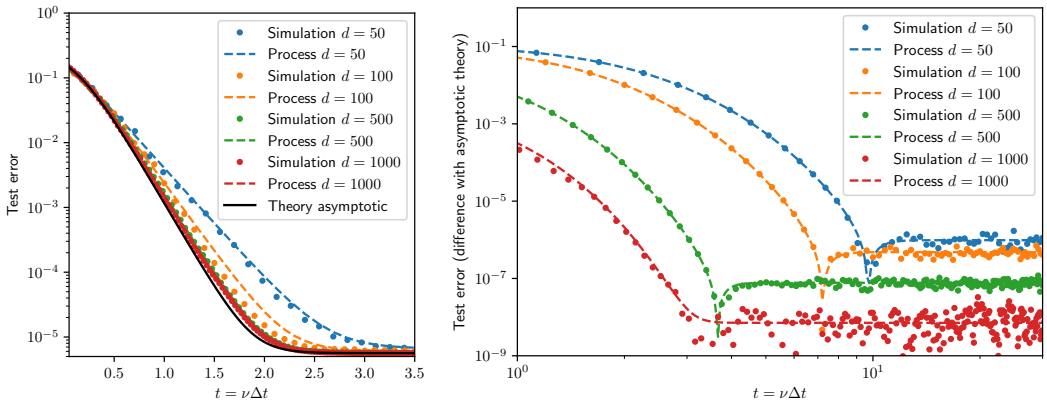


Figure D.7: **Non asymptotic corrections:** Comparison between simulations (dots), exact asymptotic solution of ODE (continuous black line), and exact solution including the subleading large-batch corrections (dashed line). As expected, as $d \to +\infty$ the simulations are getting closer and closer to the asymptotic solution; on the other hand, taking into account the batch correlations allows to have a better description of the dynamic even for small d.

Appendix E

Appendices to "Repetita Iuvant: Data Repetition Allows SGD to Learn High-Dimensional Multi-Index Functions"

E.1 Theoretical framework

E.1.1 Difference inequalities for vector-valued processes

We begin with a lemma that generalizes [BGJ21, Proposition 4.1]. Let $\Phi : \mathbb{R}^d \to \mathbb{R}^d$ be a random function, and consider the process $(\boldsymbol{u}_t)_{t\geq 0} \in \mathbb{S}^{d-1}$ given by

$$\boldsymbol{u}_{t+1} = \frac{\boldsymbol{u}_t + \gamma\Phi(\boldsymbol{u}_t)}{\left\|\boldsymbol{u}_t + \gamma\Phi(\boldsymbol{u}_t)\right\|}.$$

We make the following assumptions:

Assumption E.1. *The initial value of $\boldsymbol{u_0} \in \mathbb{S}^{d-1}$ is approximately isotropic, in the sense that there exists a constant $c > 0$ such that*

$$\mathbb{E}\left[\boldsymbol{u_0}\boldsymbol{u_0}^\top\right] \succ \frac{c}{d} I$$

Assumption E.2. *There exist constants $C, \iota > 0$ such that*

$$\sup_{\boldsymbol{u}\in\mathbb{S}^{d-1}} \left\|\mathbb{E}\left[\Phi(\boldsymbol{u})\Phi(\boldsymbol{u})^\top\right]\right\| \leq C \tag{E.1}$$

$$\sup_{\boldsymbol{u}\in\mathbb{S}^{d-1}} \mathbb{E}\left[\left\|\Phi(\boldsymbol{u})\right\|^{4+\iota}\right] \leq C d^{\frac{4+\iota}{2}} \tag{E.2}$$

Further, we have $\langle \boldsymbol{u}, \Phi(\boldsymbol{u})\rangle = 0$ a.s. for any $\boldsymbol{u} \in \mathbb{S}^{d-1}$.

We fix a vector $\boldsymbol{v}$ and consider the random process $m_t(\boldsymbol{v}) := \langle \boldsymbol{u}_t, \boldsymbol{v} \rangle$. For any $s \geq 0$, we define the stopping times

$$\tau_\theta^+(\boldsymbol{v}) = \inf\{t \geq 0 : m_t(\boldsymbol{v}) \geq \theta\} \quad \text{and} \quad \tau_\theta^-(\boldsymbol{v}) = \inf\{t \geq 0 : m_t(\boldsymbol{v}) \leq \theta\}$$

Proposition E.1. *Fix an $\alpha > 0$. There exist constants $c(\delta, \alpha), \kappa(\delta) > 0$ such that if*

$$\gamma \leq c(\delta, \alpha) d^{-1} \quad \textit{and} \quad T \leq c(\delta, \alpha) \gamma^{-2} d^{-1}, \tag{E.3}$$

then, with probability $1 - \delta$, the following holds. The initialization $m_0(\boldsymbol{v})$ satisfies

$$|m_0(\boldsymbol{v})| \geq \frac{\kappa}{\sqrt{d}}, \tag{E.4}$$

and:

- *if $m_0(v) > 0$, for any $t \leq T \wedge \tau_0^-(\boldsymbol{v})$,*

$$(1-\alpha) m_0(\boldsymbol{v}) + \gamma \sum_{s=0}^{t-1} \langle \Psi^+(\boldsymbol{u}_s), \boldsymbol{v} \rangle \leq m_t(\boldsymbol{v}) \leq (1+\alpha) m_0(\boldsymbol{v}) + \gamma \sum_{s=0}^{t-1} \langle \Psi(\boldsymbol{u}_s), \boldsymbol{v} \rangle, \tag{E.5}$$

- *if $m_0(v) < 0$, for any $t \leq T \wedge \tau_0^+(\boldsymbol{v})$,*

$$(1+\alpha) m_0(\boldsymbol{v}) + \gamma \sum_{s=0}^{t-1} \langle \Psi(\boldsymbol{u}_s), \boldsymbol{v} \rangle \leq m_t(\boldsymbol{v}) \leq (1-\alpha) m_0(\boldsymbol{v}) + \gamma \sum_{s=0}^{t-1} \langle \Psi^-(\boldsymbol{u}_s), \boldsymbol{v} \rangle, \tag{E.6}$$

where $\Psi, \Psi^\pm$ is a deterministic function equal to

$$\Psi(\boldsymbol{u}) = \mathbb{E}\left[\Phi(\boldsymbol{u})\right] \quad \textit{and} \quad \Psi^\pm(\boldsymbol{u}) = \Psi(\boldsymbol{u}) \mp \gamma \mathbb{E}\left[\left\|\Phi(\boldsymbol{u})\right\|^2\right] \boldsymbol{u}.$$

Proof. We assume for simplicity that $m_0(\boldsymbol{v}) > 0$; the case where $m_0(\boldsymbol{v}) < 0$ is treated identically. From Assumption E.1, by a simple Markov inequality, there exists a constant $\kappa(\delta) > 0$ such that

$$\mathbb{P}\left(|m_0(\boldsymbol{v})| \geq \frac{\kappa}{\sqrt{d}}\right) \geq 1 - \frac{\delta}{3}. \tag{E.7}$$

The update equation for $m_t(\boldsymbol{v})$ reads

$$m_{t+1}(\boldsymbol{v}) = \frac{m_t(\boldsymbol{v}) + \gamma \langle \Phi(\boldsymbol{u}_t), \boldsymbol{v} \rangle}{\sqrt{1 + \gamma^2 \left\|\Phi(\boldsymbol{u}_t)\right\|^2}}.$$

We then have two cases:

- either $m_t(\boldsymbol{v}) + \gamma \langle \Phi(\boldsymbol{u}_t), \boldsymbol{v} \rangle < 0$, which implies that $t+1 = \tau_0^-(\boldsymbol{v})$ and there is nothing to prove,

- or we can use the inequality $\frac{1}{\sqrt{1+x}} \geq 1 - x$, which implies that

$$\begin{aligned} m_{t+1}(\boldsymbol{v}) &\geq (m_t(\boldsymbol{v}) + \gamma\langle\Phi(\boldsymbol{u}_t), \boldsymbol{v}\rangle)(1 - \gamma^2\left\|\Phi(u_t)\right\|^2) \\ &= (1 - \gamma^2\left\|\Phi(u_t)\right\|^2)m_t + \gamma\langle\Phi(\boldsymbol{u}_t), \boldsymbol{v}\rangle - \gamma^3\langle\Phi(\boldsymbol{u}_t), \boldsymbol{v}\rangle\left\|\Phi(u_t)\right\|^2 \end{aligned}$$

By recursion, deterministically,

$$m_t(\boldsymbol{v}) \geq m_0(\boldsymbol{v}) + \gamma\sum_{s=0}^{t-1}\left(\langle\Phi(\boldsymbol{u}_s), \boldsymbol{v}\rangle - \gamma\left\|\Phi(\boldsymbol{u}_s)\right\|^2\langle\boldsymbol{u}_s, \boldsymbol{v}\rangle\right) - \sum_{s=0}^{t-1}\gamma^3\left|\langle\Phi(\boldsymbol{u}_s), \boldsymbol{v}\rangle\right|\left\|\Phi(\boldsymbol{u}_s)\right\|^2$$

Since Assumption E.2 implies Assumption B in [BGJ21], for a small enough choice of $c(\delta, \alpha)$, one has

$$\mathbb{P}\left(\sum_{t=0}^{T}\gamma^3\left|\langle\Phi(\boldsymbol{u}_t), \boldsymbol{v}\rangle\right|\left\|\Phi(\boldsymbol{u}_s)\right\|^2 \leq \frac{\alpha\kappa}{2\sqrt{d}}\right) \geq 1 - \frac{\delta}{6}.$$

Additionnally, the same assumption implies that

$$\mathrm{Var}(\langle\Phi(\boldsymbol{u}_s), \boldsymbol{v}\rangle) \leq \mathbb{E}\left[\langle\Phi(\boldsymbol{u}_s), \boldsymbol{v}\rangle^2\right] \leq C_1 \quad \text{and} \quad \mathrm{Var}(\gamma\left\|\Phi(\boldsymbol{u}_s)\right\|^2) \leq \gamma^2\mathbb{E}\left[\left\|\Phi(\boldsymbol{u}_s)\right\|^4\right] \leq c(\delta)^2C_1.$$

We can then apply Doob's martingale inequality: there exists a constant $C(\delta)$ such that with probability at least $1 - \frac{\delta}{6}$

$$\sup_{t\leq T}\left|\gamma\sum_{s=0}^{t}\left(\langle\Phi(\boldsymbol{u}_s), \boldsymbol{v}\rangle - \gamma\left\|\Phi(\boldsymbol{u}_s)\right\|^2\langle\boldsymbol{u}_s, \boldsymbol{v}\rangle - \mathbb{E}\left[\langle\Phi(\boldsymbol{u}_s), \boldsymbol{v}\rangle - \gamma\left\|\Phi(\boldsymbol{u}_s)\right\|^2\langle\boldsymbol{u}_s, \boldsymbol{v}\rangle\right]\right)\right| \leq C(\delta)\gamma\sqrt{T}.$$

The bound (E.3) on T implies that we can choose $c(\delta, \alpha)$ small enough so that

$$C(\delta)\gamma\sqrt{T} \leq \frac{C(\delta)\sqrt{c(\delta, \alpha)}}{\sqrt{d}} \leq \frac{\alpha\kappa}{2\sqrt{d}}.$$

Combining all those inequalities with a union bound, under the event of Equation (E.7),

$$\mathbb{P}\left(m_t(\boldsymbol{v}) \geq (1 - \alpha)m_0(\boldsymbol{v}) + \gamma\sum_{s=0}^{t-1}\langle\Psi^+(\boldsymbol{u}_s), \boldsymbol{v}\rangle \quad \forall t < T \wedge \tau_0^-(\boldsymbol{v})\right) \geq 1 - \frac{\delta}{3}, \quad \text{(E.8)}$$

having used that $\langle\Psi^+(\boldsymbol{u}), \boldsymbol{v}\rangle = \mathbb{E}\left[\langle\Phi(\boldsymbol{u}), \boldsymbol{v}\rangle - \gamma\left\|\Phi(\boldsymbol{u})\right\|^2\langle\boldsymbol{u}, \boldsymbol{v}\rangle\right]$.

The upper bound follows similarly: using this time the fact that for $x > 0$ we have $\frac{1}{\sqrt{1+x}} \leq 1$, we have under (E.7)

$$\mathbb{P}\left(m_t(\boldsymbol{v}) \leq (1 + \alpha)m_0(\boldsymbol{v}) + \gamma\sum_{s=0}^{t-1}\langle\Psi(\boldsymbol{u}_s), \boldsymbol{v}\rangle \quad \forall t < T \wedge \tau_0^-(\boldsymbol{v})\right) \geq 1 - \frac{\delta}{3}. \quad \text{(E.9)}$$

A union bound between the events of Equations (E.7), (E.8), (E.9) concludes the proof. □

E.1.2 Low-dimensional sufficient statistics

In [BGJ21], the function $\Psi(\boldsymbol{u})$ can be written as $\Psi(\boldsymbol{u}) = \phi(\langle \boldsymbol{u}, \boldsymbol{v}\rangle)\boldsymbol{v}$. As a result, the bounds of Proposition E.1 are actually a closed function of $m_t(\boldsymbol{v})$. We generalize this phenomenon to a larger class:

Assumption E.3. *There exists a $q > 0$ such that the function Ψ is q-approximately k-dimensional, in the sense that there exist a function $\psi : \mathbb{R}^k \to \mathbb{R}^k$ and a matrix $W^\star \in \mathbb{R}^{k\times d}$ with orthonormal rows such that for any $\boldsymbol{u} \in \mathbb{R}^d$,*

$$\left\|W^\star\Psi(\boldsymbol{u}) - \psi(W^\star\boldsymbol{u})\right\| \le \left(\left\|W^\star\boldsymbol{u}\right\| \vee \frac{1}{\sqrt{d}}\right)\cdot q.$$

By an abuse of notation, we shall sometimes view ψ as a function from $V^\star$ to itself, where $V^\star$ is the span of the rows of $W^\star$. However, Assumption E.3 ensures that all quantities related to ψ are only depending on the (oftentimes fixed) dimension k instead of d.

It turns out that in this case, we can control accurately the trajectory of $W^\star\boldsymbol{u}$, at least when f is not too flat around $\mathbf{0}$. Let $\Pi^\star = (W^\star)^\top W^\star$ be the projection operator on $V^\star$, and define the stopping time

$$\tau_\theta = \min\{t \ge 0 : \left\|\Pi^\star\boldsymbol{u}\right\| \ge \theta\}.$$

Proposition E.2. *Assume that Condition E.3 holds with q small enough, and that $\langle \boldsymbol{u}_0, \psi(\mathbf{0})\rangle > 0$ (in particular, $\psi(\mathbf{0}) \neq 0$). For any $\delta, \epsilon > 0$, there exist constants $c(\delta,\epsilon), \eta(\epsilon)$ such that if*

$$\gamma \le cd^{-1} \quad \textit{and} \quad T \le c\gamma^{-2}d^{-1}, \tag{E.10}$$

then

$$\mathbb{P}\left(\left\|\Pi^\star\boldsymbol{u}_t - \gamma t\psi(\mathbf{0})\right\| \le \sqrt{k}\eta\epsilon \quad \forall t \le T\wedge\tau_\eta\right) \ge 1-\delta. \tag{E.11}$$

Proof. Let $\boldsymbol{v}_1, \dots, \boldsymbol{v}_k$ be an orthonormal basis of $V^\star$, with $\boldsymbol{v}_1$ aligned with $\psi(\mathbf{0})$. By applying Proposition E.1 to $\delta' = \frac{\delta}{2k}$ and $\alpha = \frac{1}{2}$, with probability $1-\frac{\delta}{2}$, for any $i \in [k]$ and $t \le T\wedge\tau^{\pm}(\boldsymbol{v}_i)$, $m_t(\boldsymbol{v}_i)$ satisfies the inequality (E.5) or (E.6) (depending on the sign of $m_0(\boldsymbol{v_i})$). In particular, for $\boldsymbol{v}_1 = \psi(\mathbf{0})$, since we assumed that the initial sign is positive, we have

$$\frac{1}{2}\langle \boldsymbol{u}_0, \psi(\mathbf{0})\rangle + \gamma\sum_{s=0}^{t-1}\langle \Psi^+(\boldsymbol{u}_s), \psi(\mathbf{0})\rangle \le \langle \boldsymbol{u}_t, \psi(\mathbf{0})\rangle \le \frac{3}{2}\langle \boldsymbol{u}_0, \psi(\mathbf{0})\rangle + \gamma\sum_{s=0}^{t-1}\langle \Psi(\boldsymbol{u}_s), \psi(\mathbf{0})\rangle. \tag{E.12}$$

Define $C(\delta)$ such that $\langle \boldsymbol{u}_t, \psi(\mathbf{0})\rangle \le C(\delta)d^{-1/2}$ with probability at least $1-\frac{\delta}{2k}$. By a Taylor expansion, there exists an $\eta(\epsilon) > 0$ such that for any $\boldsymbol{v} \in \mathbb{R}^d$ with $\|\boldsymbol{v}\| \le \eta$, $\left\|\psi(\boldsymbol{v}) - \psi(\mathbf{0})\right\| \le \epsilon'$, where ϵ' will be fixed later. With Assumption E.3, this implies that for any $\boldsymbol{u} \in \mathbb{R}^d$,

$$\left\|\psi(\mathbf{0})\right\|^2 - (\epsilon' + \eta q)\left\|\psi(\mathbf{0})\right\| \le \langle \Psi(\boldsymbol{u}), \psi(\mathbf{0})\rangle \le \left\|\psi(\mathbf{0})\right\|^2 + (\epsilon' + \eta q)\left\|\psi(\mathbf{0})\right\|.$$

Further, by Jensen's inequality, $\mathbb{E}\left[\left\|\Phi(\boldsymbol{u})\right\|^2\right] \le C_1 d$ so by choosing $c(\delta,\epsilon)$ small enough, we can ensure that

$$\langle \Psi^+(\boldsymbol{u}), \psi(\mathbf{0})\rangle \ge \left\|\psi(\mathbf{0})\right\|^2 - (2\epsilon + \eta q)\left\|\psi(\mathbf{0})\right\|$$

Plugging those estimates in (E.12), we find that for $t \leq T \wedge \tau_0^-(\boldsymbol{v_1}) \wedge \tau_\eta$, and d large enough so that

$$C(\delta)d^{-1/2} \leq \frac{\eta\epsilon}{3}\left\|\psi(\mathbf{0})\right\|,$$

we have

$$\gamma t\left(\left\|\psi(\mathbf{0})\right\|^2 - (2\epsilon' + \eta q)\|\psi(\mathbf{0})\|\right) \leq \langle \boldsymbol{u}_t, \psi(\mathbf{0})\rangle \leq \frac{\eta\epsilon}{2}\|\psi(\mathbf{0})\| + \gamma t\left(\left\|\psi(\mathbf{0})\right\|^2 + (\epsilon' + \eta q)\|\psi(\mathbf{0})\|\right). \tag{E.13}$$

For small enough ϵ and q, the LHS of the above inequality is always at least $\gamma t\left\|\psi(0)\right\|^2/2$, which implies that

$$\tau_0^-(\boldsymbol{v}_1) \geq \tau_\eta \quad \text{and} \quad T \wedge \tau_\eta \leq \frac{2\eta}{\gamma\left\|\psi(\mathbf{0})\right\|} \tag{E.14}$$

Choosing ϵ' and η small enough so that

$$\frac{(2\epsilon' + \eta q)}{\left\|\psi(\mathbf{0})\right\|^2} \leq \frac{\epsilon}{2}$$

we ensure that for $t \leq T \wedge \tau_\eta$ we have

$$\left|\langle \boldsymbol{u_t} - \gamma t\psi(\mathbf{0}), \psi(\mathbf{0})\rangle\right| \leq \epsilon\eta\left\|\psi(\mathbf{0})\right\|. \tag{E.15}$$

We now move on to the other $\boldsymbol{v}_i$. For $i \geq 2$, $\boldsymbol{v}_i$ is orthogonal to $\psi(\mathbf{0})$, and this time Assumption E.3 implies

$$\left|\langle \Psi(\boldsymbol{u}), \boldsymbol{v}_i\rangle\right| \leq \epsilon + \eta q.$$

Regardless of the sign of $m_0(\boldsymbol{v}_i)$, this implies that for $t \leq T \wedge \tau_\eta$,

$$|\langle \boldsymbol{u_t}, \boldsymbol{v}_i\rangle| \leq \frac{3}{2}|\langle \boldsymbol{u}_0, \boldsymbol{v}_i\rangle| + \gamma t(\epsilon' + \eta q) \leq \frac{\eta\epsilon}{2} + \frac{2\eta(2\epsilon' + \eta q)}{\left\|\psi(\mathbf{0})\right\|},$$

having used the bound (E.14) on τ_η and bounding the initial value as above with probability $1 - \frac{\delta}{2k}$. We can always choose η, ϵ' such that the above bound is at most ϵ; since

$$\left\|\Pi_V \boldsymbol{u}_t - \gamma t\psi(\mathbf{0})\right\|^2 = \frac{\langle \boldsymbol{u_t} - \gamma t\psi(\mathbf{0}), \psi(\mathbf{0})\rangle^2}{\left\|\psi(\mathbf{0})\right\|^2} + \sum_{i=2}^{k}\langle \boldsymbol{u_t}, \boldsymbol{v}_i\rangle^2 \leq k\epsilon^2,$$

this concludes the proof. □

Proposition E.3. *Assume that $\psi(\mathbf{0}) = \mathbf{0}$, and $H_\psi := d\psi(\mathbf{0}) \neq 0$ is symmetric. Let $\bar{\lambda}$ be the largest eigenvalue of H_ψ, and assume that $\bar{\lambda} > 0$. For any $\delta, \epsilon > 0$, there exist constants $c(\delta, \epsilon), \eta(\epsilon), c'(\delta, \epsilon), C(\delta, \epsilon)$ such that if*

$$\gamma \leq cd^{-1} \quad \textit{and} \quad T \leq c\gamma^{-2}d^{-1}, \tag{E.16}$$

then

$$\mathbb{P}\left(\left\|\Pi^\star \boldsymbol{u}_t\right\| \geq c'\left\|\Pi^\star \boldsymbol{u}_0\right\| e^{\gamma(\bar{\lambda} - Ck(\epsilon+q))t} \quad \forall t \leq T \wedge \tau_\eta\right) \geq 1 - \delta. \tag{E.17}$$

Further, the mass of $\Pi^\star \boldsymbol{u}_t$ *is concentrated on the top eigenspace of* H_ψ*: there exists a* $\tilde{\lambda} < \bar{\lambda} - Ck(\epsilon + q)$ *such that on the same event,*

$$\left\|\Pi^\star \boldsymbol{u}_t - \bar{\Pi}^\star \boldsymbol{u}_t\right\| \le \left\|\Pi^\star \boldsymbol{u}_0\right\| e^{\gamma \tilde{\lambda} t} \quad \forall t \le T \wedge \tau_\eta,$$

where $\bar{\Pi}^\star$ *is the projection on the eigenspace associated to* $\bar{\lambda}$*.*

Proof. Fix $(\boldsymbol{v}_1, \dots, \boldsymbol{v}_k)$ an orthonormal basis of $V^\star$, consisting of eigenvectors of H_ψ, and let λ_i be the eigenvalue associated with $\boldsymbol{v}_i$. The largest eigenvalue of H_ψ that differs from $\bar{\lambda}$ will be denoted $\underline{\lambda}$ (we can let $\underline{\lambda} = -\infty$ if $H_\psi = \bar{\lambda} I$).

We choose an $\eta(\epsilon)$ such that for $\boldsymbol{v} \in \mathbb{R}^k$ such that $\|\boldsymbol{v}\| \le \eta$,

$$\left\|\psi(\boldsymbol{v}) - H_\psi \boldsymbol{v}\right\| \le \epsilon \|\boldsymbol{v}\| .$$

In particular, for any $i \in [k]$ and any vector $\boldsymbol{u}$, we have for small enough choice of $c(\delta, \epsilon)$

$$\langle \Psi^+(\boldsymbol{u}), \boldsymbol{v}_i \rangle \ge \lambda_i \langle \boldsymbol{v}_i, \boldsymbol{u} \rangle - (2\epsilon + q)\left\|\Pi^\star \boldsymbol{u}\right\| \quad \langle \Psi(\boldsymbol{u}), \boldsymbol{v}_i \rangle \le \lambda_i \langle \boldsymbol{v}_i, \boldsymbol{u} \rangle + (\epsilon + q)\left\|\Pi^\star \boldsymbol{u}\right\|$$

Applying Proposition E.1 to $\alpha = \epsilon$, and assuming that $m_0(\boldsymbol{v}_i) > 0$, we find that on an event with probability $1 - \delta/(2k)$, for $t \le T \wedge \tau_0^+(\boldsymbol{v_i}) \wedge \tau_\eta$,

$$\langle \boldsymbol{v}_i, \boldsymbol{u}_t \rangle \ge (1 - \epsilon) m_0(\boldsymbol{v}_i) + \gamma \sum_{s=0}^{t-1} \left(\lambda_i \langle \boldsymbol{v}_i, \boldsymbol{u}_s \rangle - (2\epsilon + q)\left\|\Pi^\star \boldsymbol{u}_s\right\|\right)$$
$$\langle \boldsymbol{v}_i, \boldsymbol{u}_t \rangle \le (1 + \epsilon) m_0(\boldsymbol{v}_i) + \gamma \sum_{s=0}^{t-1} \left(\lambda_i \langle \boldsymbol{v}_i, \boldsymbol{u}_s \rangle + (\epsilon + q)\left\|\Pi^\star \boldsymbol{u}_s\right\|\right)$$

The same argument when $m_0(\boldsymbol{v}_i) < 0$, we obtain that with probability $1 - \delta/2$, for any $t \le T \wedge \tau_\eta$ and $i \in [k]$:

$$|\langle \boldsymbol{v}_i, \boldsymbol{u}_t \rangle| \ge (1 - \epsilon)|\langle \boldsymbol{v}_i, \boldsymbol{u}_0 \rangle| + \gamma \sum_{s=0}^{t-1} \left(\lambda_i |\langle \boldsymbol{v}_i, \boldsymbol{u}_s \rangle| - (2\epsilon + q)\left\|\Pi^\star \boldsymbol{u}_s\right\|\right) \tag{E.18}$$
$$|\langle \boldsymbol{v}_i, \boldsymbol{u}_t \rangle| \le (1 + \epsilon)|\langle \boldsymbol{v}_i, \boldsymbol{u}_0 \rangle| + \gamma \sum_{s=0}^{t-1} \left(\lambda_i |\langle \boldsymbol{v}_i, \boldsymbol{u}_s \rangle| + (\epsilon + q)\left\|\Pi^\star \boldsymbol{u}_s\right\|\right) \tag{E.19}$$

Define $\bar{\Pi}^\star$ the projection on the eigenspace associated to $\bar{\lambda}$, and $\underline{\Pi}^\star$ its orthogonal projection. Summing the upper bound of (E.19) on all v_i with $\lambda_i \le \underline{\lambda}$,

$$\left\|\underline{\Pi}^\star \boldsymbol{u}_t\right\|_1 \le (1 + \epsilon)\left\|\underline{\Pi}^\star \boldsymbol{u}_0\right\|_1 + \gamma(\underline{\lambda} + k(\epsilon + q)) \sum_{s=0}^{t-1} \left\|\Pi^\star \boldsymbol{u}_s\right\|_1 ,$$

where the ℓ^1 norm $\|\cdot\|_1$ in $V^\star$ is computed w.r.t the basis $(\boldsymbol{v}_1, \dots, \boldsymbol{v}_k)$, and we used the norm bound $\|\Pi^\star \boldsymbol{u}_s\| \le \|\Pi^\star \boldsymbol{u}_s\|_1$.

Call $\mathcal{P}(t_0)$ the following condition:

$$\left\|\underline{\Pi}^\star \boldsymbol{u}_t\right\|_1 \leq C\left\|\bar{\Pi}^\star \boldsymbol{u}_t\right\|_1 \quad \text{for all } t < t_0, , \tag{E.20}$$

where C will be fixed later, and assume that $\mathcal{P}(t_0)$ holds for some $t_0 > 0$. Using the discrete Grönwall inequality, the following holds: for any $t \leq T \wedge \tau_\eta \wedge t_0$,

$$\left\|\underline{\Pi}^\star \boldsymbol{u}_t\right\|_1 \leq (1+\epsilon)\left\|\underline{\Pi}^\star \boldsymbol{u}_0\right\|_1 e^{\gamma(\underline{\lambda}+(1+C)k(\epsilon+q))t}. \tag{E.21}$$

We now sum the lower bound of (E.18) on all v_i with $\lambda_i = \bar{\lambda}$, to obtain

$$\left\|\bar{\Pi}^\star \boldsymbol{u}_t\right\|_1 \geq (1-\epsilon)\left\|\bar{\Pi}^\star \boldsymbol{u}_0\right\|_1 + \gamma \sum_{s=0}^{t-1} (\bar{\lambda} - k(2\epsilon+q))\left\|\bar{\Pi}^\star \boldsymbol{u}_s\right\|_1 - k(2\epsilon+q)\left\|\underline{\Pi}^\star \boldsymbol{u}_s\right\|_1 .$$

above bound implies that for all $t \leq t_0$,

$$\left\|\bar{\Pi}^\star \boldsymbol{u}_t\right\|_1 \geq (1-\epsilon)\left\|\bar{\Pi}^\star \boldsymbol{u}_0\right\|_1 + \gamma \sum_{s=0}^{t-1} (\bar{\lambda} - (1+C)k(2\epsilon+q))\left\|\bar{\Pi}^\star \boldsymbol{u}_s\right\|_1 .$$

By the discrete Grönwall inequality,

$$\begin{aligned}
\left\|\bar{\Pi}^\star \boldsymbol{u}_t\right\|_1 &\geq (1-\epsilon)\left\|\bar{\Pi}^\star \boldsymbol{u}_0\right\|_1 \left(1 + \gamma(\bar{\lambda} - (1+C)k(2\epsilon+q))\right)^t \\
&\geq (1-\epsilon)\left\|\bar{\Pi}^\star \boldsymbol{u}_0\right\|_1 \exp\left(\gamma(\bar{\lambda} - (1+C)k(2\epsilon+q))t - O(\gamma^2 T)\right) \\
&\geq (1-2\epsilon)\left\|\bar{\Pi}^\star \boldsymbol{u}_0\right\|_1 \exp\left(\gamma(\bar{\lambda} - (1+C)k(2\epsilon+q))t\right), \qquad \text{(E.22)}
\end{aligned}$$

where the last inequality holds for large enough d because of the bound (E.3). Now, fix a constant $r(\delta)$ such that with probability $1-\delta/2$, $\left\|\bar{\Pi}^\star \boldsymbol{u}_0\right\|_1 \leq r(\delta)\|\underline{\Pi}^\star \boldsymbol{u}_0\|_1$. Then condition $\mathcal{P}(1)$ holds as long as $C > r(\delta)$, and $\mathcal{P}(t_0)$ implies $\mathcal{P}(t_0+1)$ whenever the lower bound of eq. (E.22) is higher than the upper bound of eq. (E.21) for $t \leq t_0$. In particular, this holds whenever

$$C \geq \frac{1+\epsilon}{1-2\epsilon} r(\delta) \quad \text{and} \quad \bar{\lambda} - (1+C)k(2\epsilon+q) \geq \underline{\lambda} + (1+C)k(\epsilon+q).$$

The above condition can be satisfied if and only if

$$\frac{1+\epsilon}{1-2\epsilon} r(\delta) < \frac{\bar{\lambda} - \underline{\lambda}}{3\epsilon + 2q} - 1,$$

which can always be satisfied when ϵ, q are low enough. The proposition ensues from the remark that $\left\|\bar{\Pi}^\star \boldsymbol{u}_t\right\| \leq \|\Pi^\star \boldsymbol{u}\| \leq (1+C)\left\|\bar{\Pi}^\star \boldsymbol{u}_t\right\|$ for any $t \leq T \wedge \tau_\eta$. □

E.1.3 Bounding the hitting times

It finally remains to show how the bounds of Propositions E.2 and E.3 allow us to bound the hitting times τ_η. We summarize those in the following proposition:

Proposition E.4. *Let $\boldsymbol{u}$ be a process that satisfies Assumptions E.1-E.3 for low enough q. Then for any $\delta, \epsilon > 0$, there exist constants $\gamma_0(\delta,\epsilon), \eta(\epsilon), \kappa(\delta), K(\delta), C(\delta,\epsilon)$ such that, with probability $1-\delta$:*

(i) if $\psi(\mathbf{0}) \neq \mathbf{0}$ and $\langle \boldsymbol{u}_0, \psi(\mathbf{0})\rangle > 0$, letting $\gamma = \gamma_0 d^{-1}$,

$$\frac{\eta(1-\sqrt{k}\epsilon)}{\gamma\|\psi(\mathbf{0})\|} \leq \tau_\eta \leq \frac{\eta(1+\sqrt{k}\epsilon)}{\gamma\|\psi(\mathbf{0})\|}$$

(ii) if $\psi(\mathbf{0}) = 0$ and $d\psi(\mathbf{0})$ is symmetric and has a positive maximum eigenvalue $\bar{\lambda}$, then letting $\gamma = \gamma_0(d\log(d))^{-1}$

$$\frac{\log\left(\frac{\eta\sqrt{d}}{(1+\epsilon)K}\right)}{\gamma(\bar{\lambda}+k(\epsilon+q))} \leq \tau_\eta \leq \frac{\log\left(\frac{\eta\sqrt{d}}{\kappa c'}\right)}{\gamma(\bar{\lambda}-Ck(\epsilon+q))}$$

Further, the directions of $\Pi^\star \boldsymbol{u}_{\tau_\eta}$ concentrate, in the following sense:

(i) if $\psi(\mathbf{0}) \neq \mathbf{0}$,

$$\left\| \frac{\Pi^\star \boldsymbol{u}_{\tau_\eta}}{\eta} - \frac{\psi(\mathbf{0})}{\|\psi(\mathbf{0})\|} \right\| \leq 2\sqrt{k}\epsilon$$

(ii) if $\psi(\mathbf{0}) = \mathbf{0}$, for some $\beta < 1$,

$$\|\Pi^\star \boldsymbol{u}_{\tau_\eta} - \bar{\Pi}^\star \boldsymbol{u}_{\tau_\eta}\| \leq C(\delta,\epsilon) d^{-\frac{1-\beta}{2}},$$

where $\bar{\Pi}^\star$ is the projection on the top eigenspace of $d\psi(\mathbf{0})$.

Proof. The proof for the linear case is almost immediate. From Proposition E.2, with probability $1-\delta$, we have for all $t \leq T \wedge \tau_\eta$

$$\gamma t\|\psi(\mathbf{0})\| - \sqrt{k}\eta\epsilon \leq \|\Pi^\star \boldsymbol{u}_t\| \leq \gamma t\|\psi(\mathbf{0})\| + \sqrt{k}\eta\epsilon$$

Define

$$\bar{t}_\epsilon = \frac{\eta(1+\sqrt{k}\epsilon)}{\gamma\|\psi(\mathbf{0})\|} \quad \text{and} \quad \underline{t}_\epsilon = \frac{\eta(1-\sqrt{k}\epsilon)}{\gamma\|\psi(\mathbf{0})\|}.$$

For $t \leq \underline{t}_\epsilon$, the upper bound on $\|\Pi^\star \boldsymbol{u}_t\|$ is lower than η, which directly means that $\tau_\eta \geq \underline{t}_\epsilon$. The lower bound is higher than η for $t = \bar{t}_\epsilon$, which implies that $\tau_\eta \leq \bar{t}_\epsilon$ as long as $T \geq \bar{t}_\epsilon$. Such a T satisfies the bound (E.3) as long as

$$\frac{\eta(1+\sqrt{k}\epsilon)}{\gamma\|\psi(\mathbf{0})\|} \leq c(\delta,\epsilon)\gamma^{-2}d^{-1},$$

which is equivalent to

$$\gamma \leq \frac{c(\delta,\epsilon)\|\psi(\mathbf{0})\|}{\eta(1+\sqrt{k}\epsilon)} d^{-1},$$

which can always be ensured by decreasing $\gamma_0(\delta, \epsilon)$. Finally, we use the bound

$$\left\| \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} - \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|} \right\| \leq 2\frac{\|\boldsymbol{x} - \boldsymbol{y}\|}{\|\boldsymbol{x}\|}$$

to conclude that

$$\left\| \frac{\Pi^\star \boldsymbol{u}_{\tau_\eta}}{\eta} - \frac{\psi(\mathbf{0})}{\|\psi(\mathbf{0})\|} \right\| \leq 2\sqrt{k}\epsilon.$$

We now move on to the quadratic case. By Proposition E.3 and the same reasoning as in the proof of (E.21), we have with probability $1 - \delta$, for all $t \leq T \wedge \tau_\eta$,

$$c' \left\|\Pi^\star \boldsymbol{u}_0\right\| e^{\gamma(\bar{\lambda} - Ck(\epsilon+q))t} \leq \left\|\Pi^\star \boldsymbol{u}_t\right\| \leq (1+\epsilon)\left\|\Pi^\star \boldsymbol{u}_0\right\| e^{\gamma(\bar{\lambda} + k(\epsilon+q))t}$$

Upon redefining δ, we can assume that the "good" event implies that

$$\frac{\kappa(\delta)}{\sqrt{d}} \leq \left\|\Pi^\star \boldsymbol{u}_0\right\| \leq \frac{K(\delta)}{\sqrt{d}}$$

for constants k, K. As in the linear case, the upper bound above implies that

$$\tau_\eta \geq \frac{\log\left(\frac{\eta\sqrt{d}}{(1+\epsilon)K}\right)}{\gamma(\bar{\lambda} + k(\epsilon+q))},$$

while the lower bound implies that

$$T \wedge \tau_\eta \leq \frac{\log\left(\frac{\eta\sqrt{d}}{\kappa c'}\right)}{\gamma(\bar{\lambda} - Ck(\epsilon+q))} =: \bar{t}'_\epsilon.$$

It therefore remains to show that $\bar{t}'_\epsilon \leq T$; for large enough d and small enough ϵ, q, we can write $\bar{t}'_\epsilon \leq C'(\delta, \epsilon)\gamma^{-1}\log(d)$, and

$$C'(\delta, \epsilon)\gamma^{-1}\log(d) \leq c\gamma^{-2}d^{-1} \Leftrightarrow \gamma \leq c(C'd\log(d))^{-1}.$$

This condition can again be ensured by decreasing $\gamma_0(\delta, \epsilon)$ as needed. Finally, let

$$\beta = \frac{\tilde{\lambda}}{\bar{\lambda} - Ck(\epsilon+q)},$$

where $\tilde{\lambda}$ is the one of Proposition E.3. Then $\beta < 1$ and we have at $t = \tau_\eta$

$$\left\|\Pi^\star \boldsymbol{u}_{\tau_\eta} - \bar{\Pi}^\star \boldsymbol{u}_{\tau_\eta}\right\| \leq \frac{K}{\sqrt{d}} \exp\left(\beta \log\left(\frac{\eta\sqrt{d}}{\kappa c'}\right)\right) \leq C''(\delta, \epsilon) d^{-\frac{1-\beta}{2}}.$$

□

E.2 Proof of the Main Results

E.2.1 Proof of Theorem 5.2

We formalize here the proof sketch of Theorem 5.2. We track the dynamics of $\boldsymbol{w}_t$ through the following sufficient statistic:

$$m_t := \langle \boldsymbol{w}_t, \boldsymbol{w}^\star \rangle. \tag{E.23}$$

The dynamics of m_t fall under the framework of Section E.1, by letting

$$\Phi(\boldsymbol{w}) = -\nabla^\perp_{\boldsymbol{w}} \mathcal{L}\left(f(\boldsymbol{z}; \tilde{\boldsymbol{w}}(\rho), a_0), y\right).$$

Indeed, for the spherical gradient,

$$\nabla^\perp_{\boldsymbol{w}} \mathcal{L}(f(\boldsymbol{z}; \boldsymbol{w}, a_0), y) = a_0 y \sigma'(\langle \boldsymbol{w}, z \rangle) \cdot \left(\boldsymbol{z} - \langle \boldsymbol{z}, \boldsymbol{w} \rangle \boldsymbol{w}\right).$$

, hence plugging in the expression for $\tilde{\boldsymbol{w}}(\rho)$

$$\Phi(\boldsymbol{w}) = a_0 h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z} \rangle) \sigma'\left(\langle \boldsymbol{w}, \boldsymbol{z} \rangle + a_0 \rho h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z} \rangle) \sigma'(\langle \boldsymbol{w}, \boldsymbol{z} \rangle) \cdot \|\boldsymbol{z}\|^2\right) \left(\boldsymbol{z} - \langle \boldsymbol{z}, \boldsymbol{w} \rangle \boldsymbol{w}\right). \tag{E.24}$$

Assumptions 5.1 and 5.2 on σ and $h^\star$ then directly imply Assumptions E.1 and E.2 in the Appendix. As a result, we only need to study the quantity $\Psi(\boldsymbol{w}) = \mathbb{E}\left[\Phi(\boldsymbol{w})\right]$. We show the following:

Lemma E.1. *The function Ψ is approximately one-dimensional: there exists a function ϕ such that for any $\boldsymbol{w} \in \mathbb{S}^{d-1}$, we have*

$$|\langle \boldsymbol{w}^\star, \Psi(\boldsymbol{w}) \rangle - a_0 \phi(\boldsymbol{w}, \boldsymbol{w}^\star)| \leq \frac{C}{d}$$

Proof. Define the following quantities:

$$\Xi(x, x^\star, x^\perp) = h^\star(x^\star) \sigma'(x + a_0 \rho_0 \sigma'(x) h^\star(x^\star)) x^\perp \tag{E.25}$$

$$\phi(m) = \mathbb{E}[\Xi(x, x^\star, x^\perp)] \text{ for } \begin{pmatrix} x \\ x^\star \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} 1 & m \\ m & 1 \end{pmatrix}\right) \text{ and } x^\perp = x^\star - mx. \tag{E.26}$$

For simplicity, we let $u = \langle \boldsymbol{w}, \boldsymbol{z} \rangle + a_0 \rho_0 \sigma'(\langle \boldsymbol{w}, \boldsymbol{z} \rangle) h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z} \rangle)$. Doing a Taylor expansion of σ' in the second order, we have

$$\begin{aligned}\sigma'\left(\langle \boldsymbol{w}, \boldsymbol{z} \rangle + \rho h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z} \rangle) \sigma'(\langle \boldsymbol{w}, \boldsymbol{z} \rangle) \cdot \|\boldsymbol{z}\|^2\right) =& \sigma'(u) + \sigma''(u) \rho h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z} \rangle) \sigma'(\langle \boldsymbol{w}, \boldsymbol{z} \rangle) \cdot \left(\|\boldsymbol{z}\|^2 - d\right) \\ &+ R(u) \rho^2 h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z} \rangle)^2 \sigma'(\langle \boldsymbol{w}, \boldsymbol{z} \rangle)^2 \cdot \left(\|\boldsymbol{z}\|^2 - d\right)^2,\end{aligned}$$

where $R(u)$ is the Taylor remainder of σ' at second order around u.

The distribution of the triplet $(\langle \boldsymbol{w}, \boldsymbol{z}\rangle, \langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle, \langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle)$ is the same as the one of $(x, x^\star, x^\perp)$, hence

$$\mathbb{E}\left[\langle \Phi(\boldsymbol{z}), \boldsymbol{w}^\star\rangle\right] - a_0\phi(m) = \underbrace{\rho\mathbb{E}\left[\sigma''(u)h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle)^2\sigma'(\langle \boldsymbol{w}, \boldsymbol{z}\rangle)\cdot\left(\|\boldsymbol{z}\|^2 - d\right)\langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle\right]}_{(1)}$$
$$+ \underbrace{\rho^2\mathbb{E}\left[R(u)h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle)^3\sigma'(\langle \boldsymbol{w}, \boldsymbol{z}\rangle)^2\cdot\left(\|\boldsymbol{z}\|^2 - d\right)^2\langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle\right]}_{(2)}.$$

We first bound (2). From the Hölder inequality (since $\frac{1}{2}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}=1$),

$$(2) \le \rho^2\mathbb{E}\left[R(u)^8\right]^{\frac{1}{8}}\mathbb{E}\left[h^\star(x^\star)^8\right]^{\frac{1}{8}}\mathbb{E}\left[\sigma'(x)^8\right]^{\frac{1}{8}}\mathbb{E}\left[\langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle^8\right]^{\frac{1}{8}}\mathbb{E}\left[\left(\|\boldsymbol{z}\|^2 - d\right)^4\right]^{\frac{1}{2}}$$

Since σ is three times differentiable almost everywhere, and from the polynomial bounds of Assumption 5.4, the first four expectations are bounded by an absolute constant. The last term is the centered fourth moment of a $\chi^2(d)$ distribution, which is equal to $12d(d+4)$. As a result, using that $\rho^2 = O(d^{-2})$,

$$(2) \le Cd^{-1}.$$

For the first term, we let $\boldsymbol{z} = \boldsymbol{z}_\parallel + \boldsymbol{z}'$, where $\boldsymbol{z}_\parallel$ is the projection of $\boldsymbol{z}$ on the span of $\boldsymbol{w}$ and $\boldsymbol{w}^\star$. We can then write

$$(1) = \underbrace{\mathbb{E}\left[\sigma''(u)\rho h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle)^2\sigma'(\langle \boldsymbol{w}, \boldsymbol{z}\rangle)\cdot\left\|\boldsymbol{z}_\parallel\right\|^2\cdot\langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle\right]}_{(1')}$$
$$+ \underbrace{\mathbb{E}\left[\sigma''(u)\rho h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle)^2\sigma'(\langle \boldsymbol{w}, \boldsymbol{z}\rangle)\cdot\left(\left\|\boldsymbol{z}'\right\|^2 - d\right)\langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle\right]}_{(1'')}.$$

Up to the factor of ρ, the term $(1')$ is the expectation of a polynomially bounded function of a 2-dimensional Gaussian variable, hence

$$(1') \le C\rho \le C'd^{-1}.$$

In $(1'')$, the vector $\boldsymbol{z}'$ is independent from $\langle \boldsymbol{w}, \boldsymbol{z}\rangle$ and $\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle$, so the expectation factorizes as

$$\begin{aligned}|(1'')| &= \left|\mathbb{E}\left[\sigma''(u)\rho h^\star(\langle \boldsymbol{w}^\star, \boldsymbol{z}\rangle)^2\sigma'(\langle \boldsymbol{w}, \boldsymbol{z}\rangle)\langle \boldsymbol{w}^\star, \boldsymbol{z}^\perp\rangle\right]\cdot\mathbb{E}\left[\left(\left\|\boldsymbol{z}'\right\|^2 - d\right)\right]\right| \\ &\le C\rho|d-2-d| \le C'd^{-1},\end{aligned}$$

where we used that $\boldsymbol{z}'$ is a $(d-2)$-dimensional Gaussian variable. This concludes the proof. □

Lemma E.1 implies that Assumption E.3 holds for Ψ and any choice of q, as long as d is large enough. As a result, the proof of Theorem 5.2 hinges on the following facts about ϕ:

Proposition E.5. *For almost any choice of ρ_0 (w.r.t to the Lebesgue measure),*

- *if $\ell_p^\star = 1$, then $\phi(0) \neq 0$;*
- *if $\ell_p^\star = 2$, then $\phi'(0) \neq 0$.*

of Theorem 5.2. Our goal is to apply Proposition E.4 to the function ϕ. If $\ell_p^\star = 1$, then $\phi(0) \neq 0$ by the previous lemma, and with a uniform initialization of $\boldsymbol{w}^\star$ we have $m_0\phi(0) > 0$ with probability $1/2$. Combining this with the $(1-\delta)$ probability event of Proposition E.4 for small enough ϵ, with probability $1/2 - \delta$, we have $\tau_\eta \leq C(\delta)d$. We now move to the proof of the case where $\ell_p^\star = 2$. In this case, Proposition E.4 implies the desired result as soon as $\phi'(0) > 0$ (which corresponds in the $k = 1$ setting to the only eigenvalue of $d\phi(0)$ being positive). Since the Lebesgue measure is invariant w.r.t multiplication by ± 1, the factor $a_0\rho_0$ in the definition of $\phi'(0)$ has the same distribution as ρ_0, hence we can write $\phi'(0) = a_0'|\phi'(0)|$ where a_0' is also uniform in $\{-1, 1\}$. The event $\phi'(0) > 0$ therefore also has probability $1/2$, and we conclude as before. □

Analysis of ϕ: proof of Proposition E.5

The main ingredient here is to write $\phi(m) = \phi(m; \rho_0)$ and differentiate w.r.t ρ_0. For simplicity, we take $a_0 = 1$ in the definition of ϕ. We show the following:

Lemma E.2. *Under Assumption 5.4, the function $\phi(m; \rho_0)$ is analytic in ρ_0. Further, we have*

$$\left.\frac{\partial^k \phi(m;\rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} = \mathbb{E}\left[h^\star (x^\star)^{k+1} x^\perp \sigma^{(k+1)}(x)\sigma'(x)^k\right] \tag{E.27}$$

where $x, x^\star$ follow the same distribution as in eq. (E.26)*.*

Proof. The first statement stems from the analyticity of σ, and the fact that it is a property conserved through integration. For the second, we differentiate inside the expectation, to find

$$\frac{\partial \phi}{\partial \rho_0} = \mathbb{E}\left[\sigma'(x)h^\star(x^\star)^2\sigma''(x + \rho_0\sigma'(x)h^\star(x^\star))x^\perp\right];$$

the result follows by induction and taking $\rho_0 = 0$ at the end. □

We define the following quantities related to Assumption 5.2:

$$u_0^{(k)} = \mathbb{E}\left[\sigma^{(k)}(x)\sigma'(x)^{k-1}\right] \qquad u_1^{(k)} = \mathbb{E}\left[x\sigma^{(k)}(x)\sigma'(x)^{k-1}\right]$$
$$v_0^{(k)} = \mathbb{E}\left[x^\star h^\star(x^\star)^k\right] \qquad v_1^{(k)} = \mathbb{E}\left[[(x^\star)^2 - 1]h^\star(x^\star)^k\right]$$
$$\Delta^{(k)} = \mathbb{E}\left[h^\star(x^\star)^k\right]$$

Then, by Proposition 11.37 from [ODo14], and using $x^\perp = x^\star - mx$:

$$\begin{aligned}\left.\frac{\partial^k \phi(m;\rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} &= \mathbb{E}\left[h^\star(x^\star)^{k+1} x^\star \sigma^{(k+1)}(x)\sigma'(x)^k\right] - m\cdot\mathbb{E}\left[h^\star(x^\star)^{k+1} x\sigma^{(k+1)}(x)\sigma'(x)^k\right] \\ &= \left(u_0^{(k)}v_0^{(k)} + u_1^{(k)}(v_1^{(k)} + \Delta^{(k)})\cdot m + o(m)\right) - m\left(u_1^{(k)}\Delta^{(k)} + o(1)\right) \\ &= u_0^{(k)}v_0^{(k)} + u_1^{(k)}v_1^{(k)}\cdot m + o(m)\end{aligned}$$

This allows us to show Proposition E.5:

Proof. Assume first that $\ell_p^\star = 1$. Then $\phi(0;\rho_0)$ is an analytic function of ρ_0 with

$$\left.\frac{\partial^k \phi(0;\rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} = u_0^{(k)}v_0^{(k)}.$$

Since $\ell_p^\star = 1$, there exists a $k \in \mathbb{N}$ such that $v_0^{(k)} \neq 0$, and by Assumption 5.4 the coefficient $u_0^{(k)}$ is nonzero for every k. As a result, $\phi(0;\rho_0)$ is an analytic and non-identically zero function of ρ_0, so it is non-zero for almost every choice of ρ_0, as requested. The case $\ell_p^\star = 2$ is done in a similar way, noting that this time

$$\left.\frac{\partial^k \phi'(0;\rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} = u_1^{(k)}v_1^{(k)}.$$

□

E.2.2 Proof of Theorem 5.4

The proof of Theorem 5.4 follows in the same way as the one of Theorem 5.2. Recall the definition of Φ in eq. (E.24), and define

$$\Xi(z, \boldsymbol{z}^\star, \boldsymbol{z}^\perp) = h^\star(\boldsymbol{z}^\star)\sigma'(z + a_0\rho_0\sigma'(z)h^\star(\boldsymbol{z}^\star))\boldsymbol{z}^\perp \tag{E.28}$$

$$\psi(\boldsymbol{p}) = \mathbb{E}\left[\Xi(z, \boldsymbol{z}^\star, \boldsymbol{z}^\perp)\right] \text{ for } \begin{pmatrix} z \\ \boldsymbol{z}^\star\end{pmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{pmatrix} 1 & \boldsymbol{p} \\ \boldsymbol{p} & I_k\end{pmatrix}\right) \text{ and } \boldsymbol{z}^\perp = \boldsymbol{z}^\star - z\boldsymbol{p}. \tag{E.29}$$

Again, the variables $(z, \boldsymbol{z}^\star)$ correspond to $\langle \boldsymbol{w}, \boldsymbol{z}\rangle, W^\star\boldsymbol{z}$, and the vector $\boldsymbol{p}$ to $W^\star\boldsymbol{w}$. As before, the only difference between $\mathbb{E}\left[\Phi(\boldsymbol{w})\right]$ and $\psi(W^\star\boldsymbol{w})$ lies in replacing $\|\boldsymbol{z}\|$ with d, and the proof of Lemma E.1 implies the following: for any $\boldsymbol{w} \in \mathbb{S}^{d-1}$,

$$\left\|W^\star\Psi(\boldsymbol{w}) - \psi(W^\star\boldsymbol{w})\right\| \leq \frac{C}{d}.$$

As a result, the function Ψ is approximately k-dimensional for any choice of q as $d \to \infty$.

We can view the vector $\psi(\boldsymbol{p})$ as a random variable depending on the choice of a_0, ρ_0. Since a_0 is a sign, we will only consider the dependency of ψ in ρ_0. As before, let $u_i^{(k)}$ be the i-th Hermite coefficient of $z \mapsto \sigma^{(k+1)}(z)\sigma'(z)^k$, and $C_i^{(k)}$ the i-th Hermite tensor of $h^\star$ [Gra49] – in particular, $C_i^{(k)}$ is a tensor of order i. We have the analogous result to Lemma E.2:

Lemma E.3. *Under Assumption 5.4, the function $\psi(\boldsymbol{p}; \rho_0)$ is analytic in ρ_0, with*

$$\left.\frac{\partial^k \psi(\boldsymbol{p}; \rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} = u_0^{(k)} C_1^{(k)} + u_1^{(k)} C_2^{(k)} \boldsymbol{p} + o(\|\boldsymbol{p}\|).$$

Proof. As in Lemma E.2, we can write $\psi(\mathbf{0}) = \psi(\mathbf{0}, \rho_0)$, with

$$\left.\frac{\partial^k \psi(\boldsymbol{p}; \rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} = \mathbb{E}\left[h^\star(\boldsymbol{z}^\star)^{k+1} \boldsymbol{z}^\perp \sigma^{(k+1)}(z) \sigma'(z)^k\right].$$

From [Dan+24a, Lemma 19], we have

$$\mathbb{E}\left[h^\star(\boldsymbol{z}^\star)^{k+1} \boldsymbol{z}^\star \sigma^{(k+1)}(z) \sigma'(z)^k\right] = \sum_{i=0}^{\infty} u_{i+1}^{(k)} \langle \boldsymbol{p}^{\otimes i}, C_i^{(k)} \rangle \boldsymbol{p} + \sum_{i=0}^{\infty} u_i^{(k)} C_{i+1}^{(k)} \times_{1,\dots,i} \boldsymbol{p}^{\otimes i}.$$

Since $\boldsymbol{z}^\perp = \boldsymbol{z} - z\boldsymbol{p}$, we find

$$\left.\frac{\partial^k \psi(\boldsymbol{p}; \rho_0)}{\partial \rho_0^k}\right|_{\rho_0=0} = \sum_{i=0}^{\infty} u_i^{(k)} \left(C_{i+1}^{(k)} \times_{1,\dots,i} \boldsymbol{p}^{\otimes i} - \langle C_{i+1}^{(k)}, \boldsymbol{p}^{\otimes (i+1)} \rangle \boldsymbol{p}\right). \tag{E.30}$$

For $i \geq 0$, we have $\left\| C_{i+1}^{(k)} \times_{1,\dots,i} \boldsymbol{p}^{\otimes i} \right\| = O\left(\|\boldsymbol{p}\|^i\right)$ and $\langle C_i^{(k)}, \boldsymbol{p}^{\otimes (i)} \rangle = O\left(\|\boldsymbol{p}\|^i\right)$. The only terms in the sum which are not of order $\|\boldsymbol{p}\|^2$ are the first two terms in the left sum, which yields the statement of the Lemma. □

We are now ready to show the first part of Theorem 5.4. We begin with a characterization of $U^\star$ whenever $\ell_p^\star = 1$:

Lemma E.4. *If $\ell_p^\star = 1$, then*

$$U^\star = \operatorname{span}\left(\left\{C_1^{(k)}\right\}_{k \geq 1}\right).$$

If $\ell_p^\star = 2$, then

$$U^\star = \bigoplus_{k \geq 1} \operatorname{Im}\left(C_2^{(k)}\right).$$

Proof. By the properties of the Hermite tensors, we have for any $v \in V^\star$ and any polynomial p

$$\mathbb{E}\left[y^k H_1(\langle \boldsymbol{v}, \boldsymbol{z} \rangle)\right] = \langle \boldsymbol{v}, C_1^{(k)} \rangle$$

As a result, if $\boldsymbol{v} = \sum_{k \geq 0} v_k C_1^{(k)}$, then by letting $p(y) = \sum_{k \geq 0} a_k y^k$ we have

$$\mathbb{E}\left[p(y) H_1(\langle \boldsymbol{v}, \boldsymbol{z} \rangle)\right] = \|\boldsymbol{v}\|^2 \neq 0.$$

Conversely, if $\boldsymbol{v}$ is orthogonal to every vector $C_1^{(k)}$, then $\mathbb{E}\left[y^k H_1(\langle \boldsymbol{v}, \boldsymbol{z}\rangle)\right] = 0$ for all $k \geq 0$, which implies that $\boldsymbol{v} \notin U^\star$.

The case $\ell_p^\star = 2$ is similar, since this time

$$\mathbb{E}\left[y^k H_2(\langle \boldsymbol{v}, \boldsymbol{z}\rangle)\right] = \boldsymbol{v}^\top C_2^{(k)} \boldsymbol{v}.$$

As a result, for fixed k, the largest vector space U_k such that $\mathbb{E}\left[y^k H_2(\langle \boldsymbol{v}, \boldsymbol{z}\rangle)\right]$ for all $\boldsymbol{v} \in U_k$ is $\ker(C_2^{(k)}) = \mathrm{Im}(C_2^{(k)})^\perp$. The result ensues from $U^\star = \bigcap_{k\geq 1} U_k^\perp$. □

Using Assumption 5.4, the following corollary ensues:

Corollary E.1. *If $\ell_p^\star = 1$, then for any $\boldsymbol{v} \in U^\star$ and almost every ρ_0,*

$$\langle \boldsymbol{v}, \psi(\mathbf{0})\rangle \neq 0.$$

As a result, with probability one, the set $\{\psi(\mathbf{0}; \rho_{0,i})\}_{i\in[p]}$ spans $U^\star$ as soon as $p \geq k$.

If $\ell_p^\star = 2$, then with probability at least $1/2$ on the choice of ρ_0, a_0, $d\psi(\mathbf{0})$ is symmetric and has a strictly positive eigenvalue.

Proof. Let $\boldsymbol{v} \in U^\star$, and $k \geq 1$ such that $\langle \boldsymbol{v}, C_1^{(k)}\rangle \neq 0$. Then

$$\left.\frac{\partial^k \langle \boldsymbol{v}, \psi(\mathbf{0}; \rho_0)\rangle}{\partial \rho_0^k}\right|_{\rho_0=0} = u_0^{(k)} \langle \boldsymbol{v}, C_1^{(k)}\rangle \neq 0,$$

so the function $\rho_0 \mapsto \langle \boldsymbol{v}, \psi(\mathbf{0}; \rho_0)\rangle$ is not the zero function. Since it is also analytic, it is nonzero for almost every choice of ρ.

We now prove that if $p \leq \dim(U^\star)$, then the family $\{\psi(\mathbf{0}; \rho_{0,i})\}_{i\in[p]}$ is free, which implies our second result. Assume by recursion that $\{\psi(\mathbf{0}; \rho_{0,i})\}_{i\in[p-1]}$ is free, and let $\boldsymbol{v}_p \in U^\star$ orthogonal to this family. Then with probability one, $\langle \boldsymbol{v}_p, \psi(\mathbf{0}; \rho_{0,p})\rangle \neq 0$, so $\psi(\mathbf{0}; \rho_{0,p})$ is independent from $\{\psi(\mathbf{0}; \rho_{0,i})\}_{i\in[p-1]}$.

Finally, if $\ell_p^\star = 2$, then by symmetry of the distribution of ρ_0 we can write $\psi(\boldsymbol{p}; a_0, \rho_0) = a_0 \psi(\boldsymbol{p}; 1, \rho')$ with $\rho' = a_0 \rho_0$ chosen independently from a_0. Writing

$$d\psi(\mathbf{0}; \rho') = \sum_{k\geq 1} (\rho')^k u_1^{(k)} C_2^{(k)},$$

we can see that $d\psi(\mathbf{0})$ is a symmetric matrix, which is non-zero with probability one on ρ'. Since either $d\psi(\mathbf{0})$ or $-d\psi(\mathbf{0})$ has a positive eigenvalue, $a_0 d\psi(\mathbf{0})$ does with probability $1/2$. □

We are now ready to prove Theorem 5.4.

of Theorem 5.4. We begin with the case $\ell_p^\star = 1$. Fix a $p = p(\delta)$ to be chosen later, and let $\psi_i(\boldsymbol{p}) = \psi(\boldsymbol{p}; a_{0,i}, \rho_{0,i})$ for $i \in [p]$. With probability $1/2$ on $\boldsymbol{w}_{0,i}$, we have $\langle \boldsymbol{w}_{0,i}, \psi_i(\mathbf{0})\rangle$; let p_+ be the number of such neurons. For $p = O(\log(1/\delta))$, we have $p_+ > \max(p/4, k)$ with probability at least $1 - \frac{\delta}{2}$.

In particular, with probability one, the set $\{\psi_i(\mathbf{0})\}_{i\in[p_+]}$ spans $U^\star$. Let $\epsilon > 0$ to be chosen later, $\eta_i(\epsilon)$ adapted to ψ_i in Proposition E.4, τ_i the hitting time for η_i in the i-th neuron. Consider the time $\tau = \min_i \tau_i$; applying Proposition E.4 to $\delta' = \delta/p$, we have

$$\tau \leq \max_{i\in[p_+]} \frac{\eta_i(1+\sqrt{k}\epsilon)}{\gamma_0 \left\|\psi_i(\mathbf{0})\right\|} d \leq C(\delta) d.$$

However, we also have $\tau > c(\delta)\gamma^{-1}$, and hence from Proposition E.2, for any $i \in [p_+]$

$$\left\|\boldsymbol{w}_{t,i}\right\| \geq \gamma\tau \left\|\psi_i(\mathbf{0})\right\| - \sqrt{k}\eta_i\epsilon \geq c'(\delta)$$

for small enough ϵ. This proves the first part of the theorem. Now, consider the vectors

$$\boldsymbol{u}_i = \frac{\Pi^\star \boldsymbol{w}_{\tau,i}}{\left\|\Pi^\star \boldsymbol{w}_{\tau,i}\right\|};$$

by the same argument as in Proposition E.4, for any $i \in [p_+]$,

$$\left\| \boldsymbol{u}_i - \frac{\psi_i(\mathbf{0})}{\left\|\psi_i(\mathbf{0})\right\|} \right\| \leq \sqrt{k}\epsilon.$$

Since the $\{\psi_i(\mathbf{0})\}_{i\in[p_+]}$ span $U^\star$ with probability one, so do the $\{\boldsymbol{u}_i\}_{i\in[p_+]}$ for a small enough choice of ϵ.

For the case $\ell_p^\star = 2$, we define the p_+ "good" neurons as those with $d\psi_i(\mathbf{0})$ having a positive eigenvalue. Since this happens with probability at least $1/2$ independently for each neuron, the bounds on $p(\delta)$ from earlier still apply. For those neurons, and small enough ϵ, we define the $\eta_i(\epsilon)$ as those in Proposition E.4, and $\tau = \max_i \tau_i$. Then by Proposition E.4, we have $\tau \leq C(\delta) d \log(d)^2$ for some constant $C(\delta)$.

Now, fix a vector $i \in [p_+]$. We define the j-th "excursion" to be the j-th sequence of times $\tau_i^{(j)}, \ldots, \tau_i^{(j)} + e_i^{(j)}$ such that $\left\|\Pi^\star \boldsymbol{w}_{t,i}\right\| \geq \eta_i$. Then:

- With probability at least $1 - e^{-c\gamma}$, we have $\left\|\boldsymbol{w}_{t+1,i} - \boldsymbol{w}_{t,i}\right\| \leq C\sqrt{\gamma}$, and thus for any excursion j we have $\left\|\Pi^\star \boldsymbol{w}_{\tau_i^{(j)}+e_i^{(j)}+1,i}\right\| \geq \eta_i/2$.
- between two excursions, the results of Proposition E.3 apply, and thus the norm of $\left\|\Pi^\star \boldsymbol{w}_{t,i}\right\|$ increases exponentially.

As a result, for $\tau_i \leq t \leq \tau$, we have $\left\|\Pi^\star \boldsymbol{w}_{t,i}\right\| > \eta_i/2$, which finishes the proof of Theorem 5.4. □

E.3 Additional proofs

E.3.1 Proof of Lemma 5.1

Let σ be an analytic function which is not a polynomial. Then for any $k \geq 0$, the function $\sigma^{(k)}$ is also analytic and non-identically zero, and hence so is the function $f_n = \sigma^{(k)} \cdot (\sigma')^k$. Lemma 5.1 thus ensues from the following result:

Lemma E.5. *Let f be an analytic and non-identically zero function. Then*

$$\mathbb{E}\left[f(x+b)\right] \neq 0 \quad \textit{and} \quad \mathbb{E}\left[xf(x+b)\right] \neq 0$$

for almost every b under the Lebesgue measure.

Proof. The function $\psi(b) = \mathbb{E}\left[f(x+b)\right]$ is analytic in b, and we have

$$\psi^{(k)}(0) = \mathbb{E}\left[f^{(k)}(x)\right] = c_k u_k(f),$$

where $u_k(f)$ is the k-th Hermite coefficient of f and c_k is a nonzero absolute constant. Since f is nonzero, at least one of its Hermite coefficients is nonzero, and ψ is a non-identically zero analytic function. As a result, $\psi(b) \neq 0$ for almost every choice of b.

The other case is handled as the first one, noting that by Stein's lemma

$$\mathbb{E}\left[xf(x+b)\right] = \mathbb{E}\left[f'(x+b)\right].$$

□

The above shows that for fixed $n \in \mathbb{N}$, the set

$$\mathcal{B}_n = \left\{ b \in \mathbb{R} \,:\, \mathbb{E}\left[\sigma^{(n)}(x+b)\sigma'(x+b)^{n-1}\right] = 0 \quad \text{or} \quad \mathbb{E}\left[x\sigma^{(n)}(x+b)\sigma'(x+b)^{n-1}\right] = 0 \right\}$$

has measure 0. Since there is a countable number of such subsets, the set

$$\mathcal{B} = \bigcup_{n \in \mathbb{N}} \mathcal{B}_n$$

also has measure zero, which is equivalent to the statement of Lemma 5.1.

E.3.2 Proof of Theorem 5.3

We decompose the polynomial $h^\star$ as an even and odd part $e(x)$ and $o(x)$, with degrees d_o and d_e. We first assume that o is non-zero, and that the leading coefficient of o is positive. Then, for odd $m \geq 0$, we have

$$\mathbb{E}\left[xh^\star(x)^m\right] = \sum_{k=0}^{m} \binom{m}{k} \mathbb{E}\left[xe(x)^k o(x)^{m-k}\right]$$

Each term in the above sum is zero if k is odd, we focus on the terms where k is even. There exist constants $A, \varepsilon > 0$ such that if $|x| \geq A$,

$$|e(x)| \geq \varepsilon |x|^{d_e} \quad \text{and} \quad |o(x)| \geq \varepsilon |x|^{d_o},$$

and we let

$$B = \sup_{x \in [-A,A]} |e(x)| \vee \sup_{x \in [-A,A]} |o(x)|$$

As a result, when k is even, both $e(x)^k$ and $xo(x)^{m-k}$ are even polynomials with positive leading coefficients, so

$$\begin{aligned}\mathbb{E}\left[xe(x)^k o(x)^{m-k}\right] &= \mathbb{E}\left[xe(x)^k o(x)^{m-k}\mathbf{1}_{x\in[-A,A]}\right] + \mathbb{E}\left[xe(x)^k o(x)^{m-k}\mathbf{1}_{x\notin[-A,A]}\right] \\ &\geq \varepsilon^2 \mathbb{E}\left[x^{kd_e+(m-k)d_o+1}\mathbf{1}_{x\notin[-A,A]}\right] - AB^m\end{aligned}$$

Let $d(m,k) = kd_e + (m-k)d_o + 1$. Then,

$$\mathbb{E}\left[xe(x)^k o(x)^{m-k}\right] \geq \varepsilon^2 \mathbb{E}\left[x^{d(m,k)}\right] - A^{d(m,k)} - AB^m.$$

Going back to the sum, and with the crude bound $\binom{m}{k} \leq 2^m$,

$$\begin{aligned}\mathbb{E}\left[xh^\star(x)^m\right] &\geq \varepsilon^2 \mathbb{E}\left[x^{d(m,0)}\right] - \sum_{k=1}^{m} 2^m \left(A^{d(m,k)} + AB^m\right) \\ &\geq \mathbb{E}\left[x^{m+1}\right] - C^m\end{aligned}$$

Since the Gaussian moments grow faster than any power of m, this last expression is non-zero for a large enough choice of m.

Finally, if o is the zero polynomial, we can do the same reasoning with $e(x)^m$ to get

$$\mathbb{E}\left[(x^2-1)e(x)^m\right] = \varepsilon^2(\mathbb{E}\left[x^{md_e+2}\right] - \mathbb{E}\left[x^{md_e}\right]) - C^m = \varepsilon^2 m d_e \mathbb{E}\left[x^{md_e}\right] - C^m,$$

and we conclude as before.

E.3.3 Hardness of sign functions

We show in the appendix the following statement:

Proposition E.6. *Let $m \in \mathbb{N}$, and*

$$h^\star(\boldsymbol{x}) = \text{sign}(x_1 \dots x_m)$$

Then the function $h^\star$ has Information and Generative Exponents $\ell = \ell^\star = m$.

We first show the lower bound. For any vector $\boldsymbol{v} \in \mathbb{R}^m$, and $k < m$, $H_k(\langle \boldsymbol{v}, \boldsymbol{x}\rangle)$ is a polynomial in z of degree k, and therefore each monomial term is missing at least one variable. We show that this implies the lower bound of Proposition E.6 through the following lemma:

Lemma E.6. *Let $f : \{-1, 1\} \to \mathbb{R}$ be an arbitrary function, and $g : \mathbb{R}^m \to \mathbb{R}$ a function which is independent from x_m. Then*

$$\mathbb{E}\left[f(h^\star(\boldsymbol{x}))g(\boldsymbol{x})\right] = \frac{f(1) + f(-1)}{2}\mathbb{E}\left[g(\boldsymbol{x})\right]$$

Proof. We simply write

$$\begin{aligned}
\mathbb{E}\left[f(h^\star(\boldsymbol{x}))g(\boldsymbol{x})\right] &= \mathbb{E}\left[\mathbb{E}\left[f(h^\star(\boldsymbol{x}))g(\boldsymbol{x}) \mid x_1, \dots, x_{m-1}\right]\right] \\
&= \mathbb{E}\left[g(\boldsymbol{x})\mathbb{E}\left[f(h^\star(\boldsymbol{x})) \mid x_1, \dots, x_{m-1}\right]\right] \\
&= \mathbb{E}\left[g(\boldsymbol{x})\frac{f(1) + f(-1)}{2}\right] \\
&= \frac{f(1) + f(-1)}{2}\mathbb{E}\left[g(\boldsymbol{x})\right],
\end{aligned}$$

where at the second line we used that g only depends on $x_1, \dots, x_{m-1}$ and at the third line the addition of x_m randomly flips the sign of $h^\star$. □

Using this property on every monomial in $H_k(\langle \boldsymbol{v}, \boldsymbol{x}\rangle)$ and summing back, we have for any function f

$$\mathbb{E}\left[f(h^\star(x))H_k(\langle \boldsymbol{v}, \boldsymbol{x}\rangle)\right] = \frac{f(1) + f(-1)}{2}\mathbb{E}\left[H_k(\langle \boldsymbol{v}, \boldsymbol{x}\rangle)\right] \tag{E.31}$$

But the latter is zero when $k \geq 1$ by orthogonality of the Hermite polynomials. This shows that $\ell^\star_{\boldsymbol{v}} \geq m$ for all $\boldsymbol{v} \in \mathbb{R}^m$, and therefore $m \leq \ell^\star \leq \ell$.

For the upper bound, let $\boldsymbol{v} = \boldsymbol{e}_1 + \cdots + \boldsymbol{e_m}$. Then

$$H_m(\langle \boldsymbol{v}, \boldsymbol{x}\rangle) = m! x_1 \dots x_m + Q(\boldsymbol{x}),$$

where Q is a polynomial where each monomial has a missing variable. As a result,

$$\begin{aligned}
\mathbb{E}\left[h^\star(\boldsymbol{x})H_m(\langle \boldsymbol{v}, \boldsymbol{x}\rangle)\right] &= m!\mathbb{E}\left[h^\star(\boldsymbol{x})x_1 \dots x_m\right] \\
&= m!\mathbb{E}\left[|x_1 \dots x_m|\right] \\
&= m!\left(\sqrt{\frac{2}{\pi}}\right)^m
\end{aligned}$$

which is a non-zero value, hence $\ell^\star \leq \ell \leq m$.

E.4 Additional Numerical Investigation

In this section, we present further numerical investigation to support our theory and extend it beyond the mathematical hypotheses used in the formal proof.

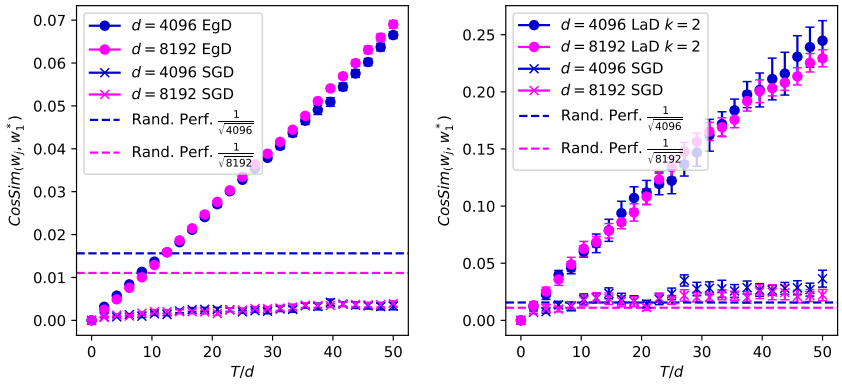


Figure E.1: example of two different algorithm with data repetitions learning an hard single-index target $h^\star(\boldsymbol{z}) = \mathrm{He}_3(z_1), \ell = 3, \ell^\star = 1$. **Left**: SAM, with $\rho_0 = 0.1, \gamma_0 = 0.01$. **Right**: 2-Lookahed with $\gamma_0 = 0.1$. See details in Appendix E.5.

E.4.1 Testing wider range of Algorithm 5.1

In the main we presented all the numerical simulation using $\rho > 0$, i.e. *ExtraGradient method-/descent*. In Figure E.1 Left, we repeat one of the experiment of Figure 5.1 in the case $\rho < 0$, the *Sharpness-Aware Minimization*. The two algorithms are based on very different principle, but in our context they behave in the same way: the reusing of the data makes high information exponent function easily learnable. The global picture for $\rho \neq 0$ is the same regardless of its sign.

Although Algorithm 5.1 includes a broad variety of algorithms, it is not the most general domain of validity of our theory. Simple algorithms such as repeating the gradient step twice on the same data before discarding it (also known as *2-lookahead optimizer* [Zha+19]) are not part of Algorithm 5.1, yet our considerations are still valid. In fact, the data repetition is sufficient to introduce correlations across two consecutive steps, whose effect builds up along the dynamics causing the network to learn hard targets. A simple illustration of this claim is available in Figure E.1 Right.

E.4.2 Extensive batch size

In this section, we want to present some evidence that our result does not change when the batch size is $1 < n_b \leq O(d^{\frac{\ell^\star}{2}})$. The formal proof we presented in the paper is valid in the case where the batch size is $n_b = 1$, but we strongly believe it is the case when training with larger batches.

In the context of the information exponent, it has been shown in [Arn+24b] that the total sample complexity does not change when using batch sizes $n_b \leq O\left(d^{\frac{\ell}{2}}\right)$. The heuristic argument for this behavior is that using a larger batch size increases the learning rate since the noise of the gradient is "more" averaged out, ultimately reducing the number of steps. The same argument should pass from information exponent to generative exponent, allowing the claim (when $n_b \leq$

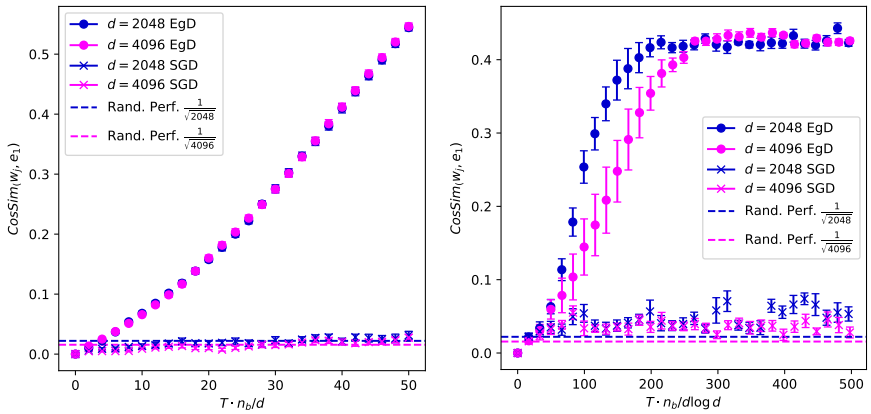


Figure E.2: Single-index recovery for large-batch training. The dashed horizontal line $\frac{1}{\sqrt{d}}$ is a visual guide to place random performance. The plots show $\sigma = \text{relu}$, $\gamma = 0.01$, $\rho = 0.1$, and the performance is computed averaging across 10 different runs. **Left** $(\ell, \ell^\star) = (3, 1)$**:** $h^\star = \text{He}_3$. **Right** $(\ell, \ell^\star) = (4, 2)$**:** $h^\star = \text{He}_4$. See details in Appendix E.5.

$$O\left(d^{\frac{\ell^\star}{2}}\right))$$

$$T \cdot n_b \sim \begin{cases} O(d^{\ell^\star - 1}) & \text{if } \ell^\star > 2 \\ O(d \log d) & \text{if } \ell^\star = 2 \\ O(d) & \text{if } \ell^\star = 1 \end{cases} . \tag{E.32}$$

Note that this drastically reduces the time steps needed to learn the teacher's directions, although the sample complexity does not change. [Dan+24b] first shows that some high-information exponent functions, such as He_3, can be learned in just a few steps with full-batch gradient descent $n_b = O(d)$. In the same work, they introduce a larger class of hard functions that cannot be recovered in $T = O(1)$, not providing any information on possible bounds on the number of time steps, nor any claim suggesting that not all the function in the class have the same sample complexity. Our theory can provide a full understanding of the phenomena:

- He_3 has generative exponent $\ell^\star = 1$, that implies we need $O(d)$ samples to weakly recover the target; when training with batch size $n_b = O(d)$, we have $T = O(1)$ as in [Dan+24b]; Figure E.2 (Left) is a numerical proof of this fact.
- He_4 has generative exponent $\ell^\star = 2$, that implies we need $O(d \log d)$ samples to weakly recover the target; when training with batch size $n_b = O(d)$, we have $T = O(\log d)$; Figure E.2 (Right) is a numerical proof of this fact.

We can push our claims beyond single-index targets and illustrate the same phenomena for multi-index functions. In Figure E.3 we test Extragradient Descent against two hard multi-index functions: once again the sample complexity is the same as the case $n_b = 1$, while the number of steps is reduced by a factor $n_b = d$.

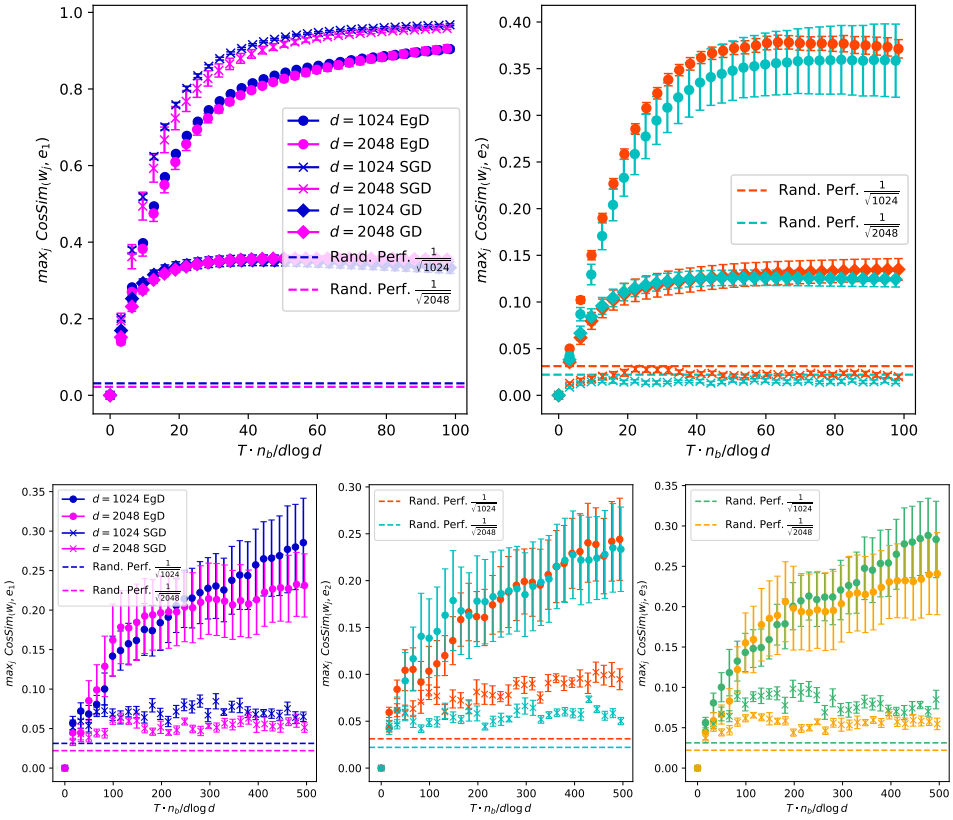


Figure E.3: Multi-index recovery for large-batch training. The dashed horizontal line $\frac{1}{\sqrt{d}}$ is a visual guide to place random performance. The plots show $\sigma = \text{relu}$. **Upper** $(\ell_{\boldsymbol{v}}, \ell^\star_{\boldsymbol{v}}) = (3, 1)$**:** $h^\star(\boldsymbol{z}) = \text{sign}(z_1 z_2)$. **Lower** $(\ell_{\boldsymbol{v}}, \ell^\star_{\boldsymbol{v}}) = (4, 2)$**:** $h^\star(\boldsymbol{z}) = z_1 z_2 z_3$. See Appendix E.5 for details.

E.5 Implementation Details

In this section, we provide all the implementation details needed to reproduce the Figures of the paper. The full implementation can be found in the repository linked in the main paper.

Unless explicitly stated, we always run the plain version of Algorithm 5.1 with squared loss, as opposed to Assumption 5.3 which requires *correlation loss* and spherical gradient. We use the reLU function as an activation. Finally, we always run simulations with $W^0 \perp W^\star$, which does not fall under Assumption 5.2. Indeed, our numerical simulations deviate from the theoretical assumptions needed for the formal proofs, but the aim here is to show that our claims are valid beyond the technical limitations of the theory and that they can allow us to understand settings closer to what is used in practice.

In the plots with a single-index target we plot the *absolute value* of the cosine similarity between the student weight $\boldsymbol{w}$ and the teacher weight $\boldsymbol{w}^\star$, averaged across N different runs (where we vary both seen samples and initial conditions). If q in the index for different runs:

$$\text{yaxis}(t) = \frac{1}{N}\sum_{q=1}^{N}\left|\text{CosSim}(\boldsymbol{w}_t^{(q)}, \boldsymbol{w}^{\star(q)})\right|,$$

where

$$\text{CosSim}(\boldsymbol{w}, \boldsymbol{w}^\star) = \frac{\boldsymbol{w}\cdot\boldsymbol{w}^\star}{\|\boldsymbol{w}\|_2\|\boldsymbol{w}^\star\|_2}.$$

In the plots with a multi-index target, we take the maximum across all the network weights

$$\text{yaxis}(t) = \frac{1}{N}\sum_{q=1}^{N}\max_{j\in[p]}\left|\text{CosSim}(\boldsymbol{w}_{j,t}^{(q)}, \boldsymbol{w}^{\star(q)})\right|.$$

E.5.1 Hyperparameters of the Figures

Here are the hyperparameters used in the figures:

- **Figure 5.1 (all 4)**: $d = 8192, \gamma_0 = 0.01, \rho_0 = 0.1, N = 40$.
- **Figure 5.3 (left)**: $p = 8, d = 2048, \gamma_0 = 0.01, \rho_0 = 0.1, N = 40, a_j^0 \sim \text{Rad}(1/2)$.
- **Figure 5.3 (center-left)**: $p = 8, d = 2048, \gamma_0 = 0.01, \rho_0 = 0.1, N = 40, a_j^0 \sim \mathcal{N}(0,1)$.
- **Figure 5.3 (center-right)**: $p = 8, d = 2048, \gamma_0 = 0.1, \rho_0 = 0.1, N = 40, a_j^0 \sim \mathcal{N}(0,1)$.
- **Figure 5.3 (right)**: $p = 8, d = 2048, \gamma_0 = 0.1, \rho_0 = 0.1, N = 20, a_j^0 \sim \mathcal{N}(0,1)$.
- **Figure 5.2 (left)**: $p = 8, d = 2048, \gamma_0 = 0.1, \rho_0 = 0.1, N = 40, a_j^0 \sim \mathcal{N}(0,1)$.
- **Figure 5.2 (right)**: $p = 4, d = 1024, \gamma_0 = 0.01, \rho_0 = 0.1, N = 10, a_j^0 \sim \mathcal{N}(0,1)$.
- **Figure 5.4**: $p = 4, d = 1024, \gamma_0 = 0.01, \rho_0 = 0.1, N = 10, a_j^0 \sim \mathcal{N}(0,1)$.

- **Figure E.1 (left)**: SAM $d = 4096, 8192, \gamma_0 = 0.01, \rho_0 = 0.1, N = 40$.
- **Figure E.1 (right)**: 2-Lookahead $d = 4096, 8192, \gamma_0 = 0.01, \rho_0 = 0.1, N = 40$.
- **Figure E.2 (all 2)**: $d = 8192, n_b = 2d, \gamma_0 = 0.1, N = 40$.
- **Figure E.3 (upper)**: $p = 8, d = 2048, n_b = 2d, \gamma_0 = 0.1, \rho_0 = 0.1, N = 40$, $a_j^0 \sim \mathcal{N}(0, 1)$.
- **Figure E.3 (lower)**: $p = 8, d = 2048, n_b = 2d, \gamma_0 = 0.1, \rho_0 = 0.1, N = 20$, $a_j^0 \sim \mathcal{N}(0, 1)$.

Appendix F

Appendices to "Asymptotics of SGD in Sequence-Single Index Models and Single-Layer Attention Networks"

F.1 Reduction from attention to sequence single-index

Let $X \in \mathbb{R}^{L\times d}$ denote a sequence of length L of d-dimensional tokens, and consider the standard dot-product attention:

$$\text{Attention}(X) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_{\text{head}}}}\right)V \tag{F.1}$$

where $Q = \left(X + \frac{P}{\sqrt{d}}\right)W_Q, K = \left(X + \frac{P}{\sqrt{d}}\right)W_K, V = \left(X + \frac{P}{\sqrt{d}}\right)W_V \in \mathbb{R}^{L\times d_{\text{head}}}$ are trainable weights known as the *query*, *key* and *value* matrices, respectively. The matrix $P \in \mathbb{R}^{L\times d}$ is the *positional encoding*, a fixed matrix needed to inject a representation of the position of the tokens in the sequence. To make the analysis tractable, [Tro+25b] considered the following simplifying assumptions:

- Key and query matrix are tied, and $d_{\text{head}} = 1$: $Q = K = \left(X + \frac{P}{\sqrt{d}}\right)\cdot \boldsymbol{w} \in \mathbb{R}^{L\times 1}$;
- Identity value matrix $V = I_L$.

Note the second assumption is mild for single-layer attention, since we do not need to learn a new representation of the sequence. Under these assumptions, Equation (F.1) reduces to:

$$\text{TiedAttention}(X) = \text{softmax}\left(\left(X + \frac{P}{\sqrt{d}}\right)\boldsymbol{w}\boldsymbol{w}^\top\left(X + \frac{P}{\sqrt{d}}\right)^\top\right) \tag{F.2}$$

This is a map from sequences $X \in \mathbb{R}^{L\times d}$ to $L \times L$ matrices. Further adding the reduction map $\Pi : \mathbb{R}^{L\times L} \to \mathbb{R}^{k_{\text{out}}}$, we get the model in Equation (6.1). Finally, to get the reduction to a

sequence single-index model, it suffices to consider $P = 0$ and the map on real-valued sequences $\boldsymbol{s} \in \mathbb{R}^L$:

$$g(\boldsymbol{s}) = \Pi\left[\text{softmax}\left(\boldsymbol{s}\boldsymbol{s}^\top\right)\right] \tag{F.3}$$

F.2 Mathematical preliminaries and notations

F.2.1 Tensors

We consider tensors as multidimensional arrays: a tensor T of order k and dimensions $(d_1, \dots, d_k)$ is simply an element of $\mathbb{R}^{d_1 \times \cdots \times d_k}$. Its elements are denoted by $T_{i_1 \dots i_k}$, where $i_\ell \in [d_\ell]$. The scalar product between two tensors with same dimensions is defined as

$$\langle T, T \rangle = \sum_{i_1, \dots, i_k} T_{i_1 \dots i_k} T'_{i_1 \dots i_k}.$$

We say that a tensor is *symmetric* if all its dimensions are equal and for any index $(i_1, \dots, i_k)$ and permutation $\sigma \in \mathfrak{S}_k$,

$$T_{i_1 \dots i_k} = T_{i_{\sigma(1)} \dots i_{\sigma(k)}}.$$

We shall need two operations on tensors: the first is the *tensor product*, that turns two tensors of order k, ℓ into a tensor of order $k + \ell$ defined as

$$(T \otimes T')_{i_1 \dots i_{k+\ell}} = T_{i_1 \dots i_k} T'_{i_{k+1} \dots i_{k+\ell}}.$$

The second is the *tensor-matrix* contraction: given a tensor T of order k, an index ℓ and a matrix M of size $d_\ell \times d'\ell$, the tensor $T \times_\ell M$ is defined as

$$(T \times_\ell M)_{i_1 \dots i'_\ell \dots i_k} = \sum_{i_\ell} T_{i_1 \dots i_\ell \dots i_k} M_{i_\ell i'_\ell}$$

Given k matrices $M^{(1)}, \dots, M^{(k)}$, we will use the shorthand

$$T \times (M^{(1)}, \dots, M^{(k)}) = T \times_1 M^{(1)} \cdots \times_k M^{(k)}$$

Immediate properties of those operations are gathered in the following lemma:

Lemma F.1. *The operation $\times$ is associative: if T is a tensor and*

$$(M^{(1)}, \dots, M^{(k)}), (N^{(1)}, \dots, N^{(k)})$$

are matrices with compatible dimensions,

$$\left(T \times (M^{(1)}, \dots, M^{(k)})\right) \times (N^{(1)}, \dots, N^{(k)}) = T \times (M^{(1)} N^{(1)}, \dots, M^{(k)} M^{(k)})$$

Let $T \in \mathbb{R}^{d_1 \times \cdots \times d_k}$ and $(\boldsymbol{x}_1, \dots, \boldsymbol{x}_k) \in \mathbb{R}^{d_1} \times \cdots \times \mathbb{R}^{d_k}$. Then

$$\langle T, \boldsymbol{x}_1 \otimes \cdots \otimes \boldsymbol{x}_k \rangle = T \times (\boldsymbol{x}_1, \dots, \boldsymbol{x}_k).$$

Odeco tensors Since tensors of order $k \geq 3$ are sometimes hard to handle, we work with a restricted class. We say that a symmetric tensor T is *odeco* (short for *orthogonally decomposable*, see [Rob16]) if there exist real numbers $\lambda_1, \ldots, \lambda_r$ and orthogonal vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$ such that

$$T = \sum_{i=1}^{r} \lambda_i v_i^{\otimes k}.$$

In particular, all tensors of order 1 (with $r = 1$) and 2 (with r equal to the rank of T) are odeco.

F.2.2 Hermite Polynomials

In this section we provide our definition of Hermite polynomials, which are used in the construction of the Hermite basis, both for the one-dimensional and the multidimensional case.

Gaussian measure and Gaussian ℓ^2 space We define the Gaussian density in p dimensions

$$\omega_p(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}} \exp\left(-\frac{\|\boldsymbol{x}\|^2}{2}\right),$$

and $\mathrm{d}\omega_p(\boldsymbol{x}) = \omega_p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$. This measure defines a space $\ell^2(\omega_p)$ of functions f satisfying

$$\|f\|_\omega := \int f(\boldsymbol{x})^2 \mathrm{d}\omega_p(\boldsymbol{x}) < \infty;$$

it is a Hilbert space w.r.t the scalar product

$$\langle f, g\rangle_\omega = \int f(\boldsymbol{x}) g(\boldsymbol{x}) \mathrm{d}\omega_p(\boldsymbol{x}).$$

Hermite polynomials and tensors We follow the conventions of [Gra49]. Define the k-th Hermite tensor $\mathcal{H}_k$ as

$$\mathcal{H}_k = \frac{(-1)^k}{\omega_p} \nabla^k \omega_p,$$

where ∇^k is the k-th order derivative. This results in a k-th order symmetric tensor of size $p \times \cdots \times p$. The Hermite tensors are orthogonal, in the sense that

$$\langle (\mathcal{H}_k)_{i_1 \ldots i_k}, (\mathcal{H}_\ell)_{j_1 \ldots j_\ell} \rangle_\omega \neq 0 \text{ if and only if } k = \ell \text{ and } (i_1, \ldots, i_k) \text{ is a permutation of } (j_1, \ldots, j_\ell).$$

When $p = 1$, all Hermite tensors are scalars, and we get the usual Hermite polynomials:

$$\mathrm{He}_0(x) = 1, \tag{F.4}$$
$$\mathrm{He}_1(x) = x, \tag{F.5}$$
$$\mathrm{He}_2(x) = x^2 - 1, \tag{F.6}$$
$$\mathrm{He}_3(x) = x^3 - 3x, \tag{F.7}$$
$$\mathrm{He}_4(x) = x^4 - 6x^2 + 3. \tag{F.8}$$

Hermite expansion The orthogonality properties of the Hermite tensors imply the following theorem:

Theorem F.1. *Let $f \in \ell^2(\omega_p)$. There exist a unique sequence of coefficients $\big(C_k(f)\big)_{k\geq 0}$ such that $C_k(f)$ is a tensor of order k and*

$$f = \sum_{k\geq 0} \langle C_k(f), \mathcal{H}_k \rangle. \tag{F.9}$$

Those coefficients are given by the following formula:

$$C_k(f) = \frac{1}{k!} \int f(\boldsymbol{x}) \mathcal{H}_k(\boldsymbol{x}) d\omega_p(\boldsymbol{x}).$$

Further, the scalar product $\langle \cdot, \cdot \rangle_\omega$ can be written as

$$\langle f, g \rangle_\omega = \sum_{k\geq 0} k! \, \langle C_k(f), C_k(g) \rangle$$

The proof of this theorem can be found in [Gra49]. The identity (F.9) is called the *Hermite expansion* of f, and the $C_k(f)$ are its *Hermite coefficients*.

Finally, by invariance of the Gaussian distribution through orthogonal transformation, the following holds:

Lemma F.2. *Let $g : \mathbb{R}^p \to \mathbb{R}$, and $W \in \mathbb{R}^{p\times q}$ be a matrix satisfying $WW^\top = I_p$. Let $f(\boldsymbol{x}) = g(W\boldsymbol{x})$. Then*

$$C_k(f) = C_k(g) \times (W, \dots, W).$$

When $p = 1$ and $\boldsymbol{w}$ is a single vector, we get

$$C_k(f) = c_k(g) \boldsymbol{w}^{\otimes k}.$$

This gives rise to a link between odeco tensors and separable functions:

Lemma F.3. *Let $g : \mathbb{R}^\ell \to \mathbb{R}$ be a separable function, such that*

$$g(\boldsymbol{z}) = \sum_i g_i(z_i),$$

$W \in \mathbb{R}^{\ell\times q}$ an orthogonal matrix, and let $f(\boldsymbol{x}) = g(W\boldsymbol{x})$. Then

$$C_k(f) = \sum_{i=1}^{\ell} c_k(g_i) \boldsymbol{w}_i^{\otimes k},$$

and in particular every Hermite coefficient of f is odeco.

F.3 Formalization and proofs

F.3.1 Preliminaries

We consider the following approximation of the SGD dynamics:

$$\frac{d\boldsymbol{w}}{dt} = -\mathbb{E} * \nabla^{\perp}_{\boldsymbol{w}} \mathcal{L}(X, y, f_{\boldsymbol{w}}) - \frac{\gamma}{2}\mathbb{E} * \left\| \nabla^{\perp}_{\boldsymbol{w}} \mathcal{L}(X, y, f_{\boldsymbol{w}}) \right\|^2 \boldsymbol{w} \tag{F.10}$$

The results of [BGJ21] (when m is a scalar) and [Arn+24c] (when $\boldsymbol{m}$ is a vector) imply the following:

Theorem F.2. *Let t_η be the weak recovery time of the ODE* (F.10)*, with the same initial conditions as the process* (6.2)*, and let ν_η^{weak} be the number of gradient steps of Definition 6.1. Then for small enough η and any $\delta > 0$, there exist constants $c(\delta), C(\delta)$ such that if*

$$\gamma = c(\delta)(dt_\eta)^{-1}, \tag{F.11}$$

then with probability at least $1 - \delta$

$$\nu_\eta^{\text{weak}} \leq C(\delta) dt_\eta^2.$$

On the other hand, for any number of steps $\nu \leq C(\delta)dt_\eta^2$, if $\gamma \leq c(\delta)(d\nu)^{-1/2}$, then with probability $1 - \delta$

$$\nu_\eta^{\text{weak}} \geq c(\delta)\nu$$

When γ does not satisfy the bound (F.11), we cannot show a strong enough concentration around the deterministic ODE dynamics, and hence directly showing non-convergence of (6.2) is difficult. However, as we shall see in the proof, above this value of γ the inhibitive term in (F.10) dominates at initialization, and hence the ODE dynamics stay trapped around zero overlap with the target subspace. For this reason, we shall consider (in line with [BGJ21]) that the sample complexity cannot be improved by increasing γ above the bound (F.11).

Remark F.1. *When γ is instead fixed below the value* (F.11)*, the hitting time of the dynamics 6.2 is instead given by*

$$\nu_\eta^{\text{weak}}(\gamma) \asymp \gamma^{-1} t_\eta.$$

We shall also assume that $m_0 > 0$, and that the coefficients appearing in the gain expression in Theorem 6.2 are all non-negative. As mentioned in [BGJ21; Arn+24b; Arn+24c], this condition can be ensured with probability $1/2$ by randomly setting the learning rate to $\pm\gamma$ with equal probability.

F.3.2 An ODE for overlap evolution

We begin by computing the expectation of the gradient term:

Lemma F.4. *In the tied case, we have when* $\|\boldsymbol{w}\| = 1$

$$\mathbb{E} * \mathcal{L}(X, y, f_{\textit{tied}}) = \mathbb{E}y^2 - 2\sum_{k\geq 0} c_k(\sigma)C_k(g) \times (\mathbf{1}_L, \dots, \mathbf{1}_L)m^k + \|\sigma\|_\omega .$$

In the untied case, we have instead

$$\mathbb{E} * \mathcal{L}(X, y, f_{\textit{untied}}) = \mathbb{E}y^2 - 2\sum_{k\geq 0} c_k(\sigma)C_k(g) \times (\boldsymbol{m}, \dots, \boldsymbol{m}) + \|\sigma\|_\omega .$$

Proof. Recall that $\mathcal{L}(X, y, f) = (y - f(X))^2 = y^2 - 2yf(X) + f(X)^2$. For simplicity, define

$$\begin{aligned}
\tilde{\boldsymbol{w}}_{\text{tied}} &= (\boldsymbol{w}\, \boldsymbol{w} \dots \boldsymbol{w}) \in \mathbb{R}^{dL}, \\
\tilde{\boldsymbol{w}}_{\text{untied}} &= (\boldsymbol{w}_1 \dots \boldsymbol{w}_L) \in \mathbb{R}^{dL}, \text{ and} \\
\tilde{W}^\star &= \begin{pmatrix} \boldsymbol{w}_1^\star & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{w}_2^\star & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{w}_L^\star \end{pmatrix} \in \mathbb{R}^{L \times dL}
\end{aligned}$$

Then, for $\square \in \{\text{tied}, \text{untied}\}$, we have

$$f_\square(X) = \sigma\left(\frac{\langle \tilde{\boldsymbol{w}}_\square, \text{flatten}(X)\rangle}{\sqrt{L}}\right) \quad \text{and} \quad y(X) = g(\tilde{W}^\star \cdot \text{flatten}(X)).$$

Since $\frac{\text{flatten}(X)}{\sqrt{L}}$ is a standard normal vector, and both $\frac{\tilde{\boldsymbol{w}}_\square}{\sqrt{L}}$ and $\frac{\tilde{W}^\star}{\sqrt{L}}$ are orthogonal matrices, we can use the Hermite expansion properties to find

$$\begin{aligned}
\mathbb{E} * f_\square(X)y(X) &= \sum_{k\geq 0} \langle C_k(f_\square), C_k(y)\rangle \\
&= \sum_{k\geq 0} \langle c_k(\sigma)\tilde{\boldsymbol{w}}_\square^{\otimes k}, C_k(g) \times (\tilde{W}^\star, \dots, W^\star)\rangle \\
&= \sum_{k\geq 0} c_k(\sigma)C_k(g) \times (\tilde{W}^\star\tilde{\boldsymbol{w}}_\square, \dots, \tilde{W}^\star\tilde{\boldsymbol{w}}_\square).
\end{aligned}$$

In the tied case, we have $\tilde{W}^\star\tilde{\boldsymbol{w}}_{\text{tied}} = m\mathbf{1}_L$, while in the untied case $\tilde{W}^\star\tilde{\boldsymbol{w}}_{\text{untied}} = \boldsymbol{m}$. For the last term, we have in both cases $\frac{\langle \tilde{\boldsymbol{w}}_\square, \text{flatten}(X)\rangle}{\sqrt{L}} \sim \mathcal{N}(0, 1)$ whenever $\|\tilde{\boldsymbol{w}}_\square\|^2 = L$, and hence

$$\mathbb{E}f_\square(X)^2 = \|\sigma\|_\omega .$$

This ends the proof. □

When $\|\boldsymbol{w}\|$ (resp. $\|\boldsymbol{w}_i\|$ is different from one, the expressions of Lemma F.4 depend on $q = \|\boldsymbol{w}\|^2$ (resp. $q_i = \|\boldsymbol{w}_i\|^2$). However, we can write

$$\nabla_{\boldsymbol{w}} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) = \frac{\partial}{\partial m} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) \boldsymbol{w}^\star + 2 \frac{\partial}{\partial q} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) \boldsymbol{w}.$$

As a result, we have

$$\nabla^\perp_{\boldsymbol{w}} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) = \left(\frac{\partial}{\partial m} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) \right) (\boldsymbol{w}^\star - m \boldsymbol{w}).$$

The same holds for the untied case:

$$\nabla^\perp_{\boldsymbol{w}} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) = \left(\frac{\partial}{\partial m_i} \mathbb{E}\mathcal{L}(X, y, f_{\text{tied}}) \right) (\boldsymbol{w}^\star - m_i \boldsymbol{w}_i)$$

We arrive at the following result:

Proposition F.1. *Define the drift functions*

$$\phi_{\textit{tied}}(m) = 2(1 - m^2) \sum_{k \geq 0} k c_k(\sigma) C_k(g) \times (\mathbf{1}_L, \dots, \mathbf{1}_L) m^{k-1}$$
$$\phi_{\textit{untied}}(\boldsymbol{m}) = 2(1 - \boldsymbol{m} \circ \boldsymbol{m}) \circ \sum_{k \geq 0} c_k(\sigma) C_k(g) \times (I_L, \boldsymbol{m}, \dots, \boldsymbol{m}).$$

Then the tied and untied overlaps satisfy the following ODEs:

$$\frac{dm}{dt} = \phi_{\textit{tied}}(m) - \frac{\gamma}{2} \mathbb{E} * \left\| \nabla^\perp_{\boldsymbol{w}} \mathcal{L}(X, y, f_{\boldsymbol{w}}) \right\|^2 m \tag{F.12}$$
$$\frac{dm}{dt} = \phi_{\textit{untied}}(\boldsymbol{m}) - \frac{\gamma}{2} \mathbb{E} * \left\| \nabla^\perp_{\boldsymbol{w}} \mathcal{L}(X, y, f_{\boldsymbol{w}}) \right\|^2 \boldsymbol{m} \tag{F.13}$$

F.3.3 Controlling the gradient norm

We turn our attention to the inhibitive terms in Proposition F.1. We show the following:

Lemma F.5. *There exist an $\eta > 0$ and two constants c, C such that if $m < \eta$ (resp. $\|m\| \leq \eta$),*

$$cd \leq \mathbb{E} * \left\| \nabla^\perp_{\boldsymbol{w}} \mathcal{L}(X, y, f_{\boldsymbol{w}}) \right\|^2 \leq Cd$$

Proof. We only treat the tied case; the untied one is done similarly. Differentiating the loss w.r.t $\boldsymbol{w}$, we find

$$\nabla_{\boldsymbol{w}} \mathcal{L}(X, y, f) = 2\sigma' \left(\frac{\langle \boldsymbol{w}_{\text{tied}}, \text{flatten}(X) \rangle}{\sqrt{L}} \right) \left(\sigma \left(\frac{\langle \boldsymbol{w}_{\text{tied}}, \text{flatten}(X) \rangle}{\sqrt{L}} \right) - y(X) \right) \cdot \frac{\sum_i \boldsymbol{x}_i}{\sqrt{L}}$$

We write $\boldsymbol{x}_i = \boldsymbol{x}_i^{\|} + \boldsymbol{x}_i^{\perp}$, where $\boldsymbol{x}_i^{\|}$ is the projection on $\boldsymbol{x}_i$ on the subspace spanned by $\boldsymbol{w}$ and $\boldsymbol{w}^\star$. Then

$$\nabla_{\boldsymbol{w}}^{\perp}\mathcal{L}(X,y,f) = 2\sigma'\left(\frac{\langle \boldsymbol{w}_{\text{tied}}, \text{flatten}(X)\rangle}{\sqrt{L}}\right)\left(\sigma'\left(\frac{\langle \boldsymbol{w}_{\text{tied}}, \text{flatten}(X)\rangle}{\sqrt{L}}\right) - y(X)\right)\cdot \frac{\sum_i (I - \boldsymbol{w}\boldsymbol{w}^\top)\boldsymbol{x}_i^{\|} + \sum_i \boldsymbol{x}_i^{\perp}}{\sqrt{L}}$$

Importantly, the vector $x_i^{\perp}$ is independent from any of the prefactors, and has squared norm $d-2$, while the first vector is a fixed-dimensional Gaussian. As a result, we have

$$\mathbb{E} * \left\|\nabla_{\boldsymbol{w}}^{\perp}\mathcal{L}(X,y,f)\right\|^2 = \mathbb{E} * f_{\boldsymbol{w}}(X)^2(y(X) - f_{\boldsymbol{w}}(X))^2(d-2) + O(1)$$

It remains to show that the expectation above is bounded away from zero. Assuming that the labels are centered for simplicity, when $m = 0$ the expectation simplifies to

$$L_0 = \mathbb{E} * f_{\boldsymbol{w}}(X)^2 \mathbb{E} * y(X)^2 + \mathbb{E} * f_{\boldsymbol{w}}(X)^4 > 0.$$

By continuity, we can choose $\eta > 0$ such that if $|m| \le \eta$

$$\frac{L_0}{2} \le \mathbb{E} * f_{\boldsymbol{w}}(X)^2(y(X) - f_{\boldsymbol{w}}(X))^2 \le 2L_0,$$

which ends the proof. □

F.3.4 Hitting time for the tied dynamics

We are now ready to prove Theorem 6.1 (as well as part of Theorem 6.2). In light of Theorem F.2, it suffices to compute the hitting time t_η of the tied dynamics (F.12). Since $\phi_{\text{tied}}(m) = 2C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)m^{\text{SIE}-1} + O(m^{\text{SIE}})$, there exists an $\eta_{\text{tied}} > 0$ such that if $m < \eta_{\text{tied}}$,

$$c_{\text{SIE}}(\sigma)C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)m^{\text{SIE}-1} < \phi_{\text{tied}}(m) < 3c_{\text{SIE}}(\sigma)C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)m^{\text{SIE}-1}$$

Combining this with the bound of Lemma F.5, we find that whenever $m(t) \le \eta$ for some constant $c > 0$,

$$\frac{dm}{dt} \ge c \cdot C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)m^{\text{SIE}-1} - C\gamma dm$$

For any $\delta > 0$, there exists a constant $\kappa(\delta)$ such that with probability at least $1 - \delta$,

$$m(0) \ge \frac{\kappa(\delta)}{\sqrt{d}}.$$

As a result, when $\gamma \le \kappa(\delta)^{\text{SIE}-1}C^{-1}d^{-\frac{\text{SIE}}{2}+1}$, then for $0 \le t \le t_\eta$

$$\frac{dm}{dt} \ge c' \cdot C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)m^{\text{SIE}-1}$$

This implies:

- when $\text{SIE} = 1$,

$$m(t) \ge m_0 + \frac{C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)}{2}t, \quad \text{hence} \quad t_\eta \le \frac{C'\eta}{C_{\text{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)};$$

- when $\mathrm{SIE} = 2$,

$$m(t) \geq m_0 \exp\left(\frac{C_{\mathrm{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)}{2} t\right), \quad \text{hence} \quad t_\eta \leq \frac{C' \ln(d)}{C_{\mathrm{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)};$$

- when $\mathrm{SIE} \geq 3$,

$$m(t) \geq \left(m(0)^{2-\mathrm{SIE}} - \frac{C_{\mathrm{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)}{2} (\mathrm{SIE} - 2) t\right)^{-\frac{1}{\mathrm{SIE}-2}},$$

hence

$$t_\eta \leq \frac{C' d^{\frac{\mathrm{SIE}}{2}-1}}{C_{\mathrm{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L)}.$$

The bound on γ above is always weaker (up to constant factors) than the bound $\gamma \leq C(dt_\eta)^{-1}$ of Theorem F.2, and hence we always have

$$\nu_\eta^{\mathrm{weak}} \leq C d t_\eta^2,$$

which corresponds to the statement of Theorem 6.1.

On the other hand, for any $t \geq 0$ we have

$$\frac{dm}{dt} \leq 3 C_{\mathrm{SIE}} \times (\mathbf{1}_L, \dots, \mathbf{1}_L) m^{\mathrm{SIE}-1},$$

which by the same reasoning implies that the bounds on t_η that we obtained are sharp up to constants.

F.3.5 Hitting time for untied dynamics

We are now ready to finish the proof of Theorem 6.2. Since

$$\phi_{\mathrm{untied}}(\boldsymbol{m}) = 2 c_{\mathrm{SIE}}(\sigma) C_{\mathrm{SIE}} \times (I_L, \boldsymbol{m}, \dots, \boldsymbol{m}) + O(\|\boldsymbol{m}\|^{\mathrm{SIE}}),$$

for small enough $\eta > 0$ we have for $t \leq t_\eta$

$$\frac{d\|\boldsymbol{m}\|}{dt} \leq C \cdot C_{\mathrm{SIE}} \times (\boldsymbol{m}, \dots, \boldsymbol{m}) + \|C_{\mathrm{SIE}}\| \|m\|^{\mathrm{SIE}-1} \leq (C+1) \|C_{\mathrm{SIE}}\| \cdot \|\boldsymbol{m}\|^{\mathrm{SIE}-1}.$$

Recall that the hitting time $t_{\eta,\mathrm{untied}}$ corresponds to $\|m\|$ hitting the threshold $\eta\sqrt{L}$. As a result, since the dependency in η is of leading order only for $\mathrm{SIE} = 1$, we find

$$t_{\eta,\mathrm{untied}} \geq \frac{c}{\|C_{\mathrm{SIE}}\|} \begin{cases} \eta\sqrt{L} & \text{if } \mathrm{SIE} = 1 \\ \log(d) & \text{if } \mathrm{SIE} = 2 \\ d^{\frac{\mathrm{SIE}}{2}-1} & \text{if } \mathrm{SIE} \geq 3 \end{cases}$$

Since the gain satisfies

$$\text{gain} \asymp \left(\frac{t_{\eta,\text{untied}}}{t_\eta} \right)^2,$$

this proves the first part of Theorem 6.2.

For the second part, assume that C_{SIE} is *odeco*, hence there exists $\lambda_1, \dots, \lambda_L$ and orthogonal vectors $\boldsymbol{v}_1, \dots, \boldsymbol{v}_L$ such that

$$C_{\text{SIE}} = \sum_{i=1}^{L} \lambda_i \boldsymbol{v}_i^{\otimes \text{SIE}}.$$

We assume that the λ_i are ordered by absolute value, so that $\|C_{\text{SIE}}\| = |\lambda_1|$ Letting $m^{(1)} = \langle \boldsymbol{m}, \boldsymbol{v_1} \rangle$, we have

$$\langle \boldsymbol{v}_1, C_{\text{SIE}} \times (I_L, \boldsymbol{m}, \dots, \boldsymbol{m}) \rangle = C_{\text{SIE}} \times (\boldsymbol{v}_1, \boldsymbol{m}, \dots, \boldsymbol{m}) = \lambda_1 (m^{(1)})^{\text{SIE}-1}$$

For small enough η, this implies that

$$\frac{dm^{(1)}}{dt} \geq c \|C_{\text{SIE}}\| \, (m^{(1)})^{\text{SIE}-1}$$

Since $\|m\| \geq m^{(1)}$ by the Cauchy-Schwarz inequality, the hitting time $t_{\eta,\text{untied}}$ is at most that of $m^{(1)}$, and hence

$$t_{\eta,\text{untied}} \leq \frac{C}{\|C_{\text{SIE}}\|} \begin{cases} \eta\sqrt{L} & \text{ifSIE} = 1 \\ \log(d) & \text{ifSIE} = 2 \\ d^{\frac{\text{SIE}}{2}-1} & \text{ifSIE} \geq 3 \end{cases}$$

This closes the upper bound of Theorem 6.2.

F.3.6 The effect of positional encoding on the flatness

We conclude with the proof of Lemma 6.1.

Proof. Let $R^{\text{new}}(\boldsymbol{e}, m)$ and $R(m)$ be the reduced population losses of Equation (6.9) for the model with and without positional encoding, respectively, so that $R(m) = R^{\text{new}}(\mathbf{0}, m)$. Then

$$\frac{\partial^k R^{\text{new}}}{\partial m^k}(\mathbf{0}, 0) = \frac{d^k R}{dm^k}(0),$$

and hence $\nabla^k R(0) \neq 0$ implies that $\nabla^k R^{\text{new}}(\mathbf{0}, 0) \neq 0$. The first non-vanishing derivative of R^{new} at the origin therefore appears at an order no larger than that of R, which is exactly $\kappa(f_{\boldsymbol{w}}^{\text{new}}, \boldsymbol{f}_{\boldsymbol{w}_\star}^{\text{SSI}}) \leq \kappa(f_{\boldsymbol{w}}, \boldsymbol{f}_{\boldsymbol{w}_\star}^{\text{SSI}})$. □

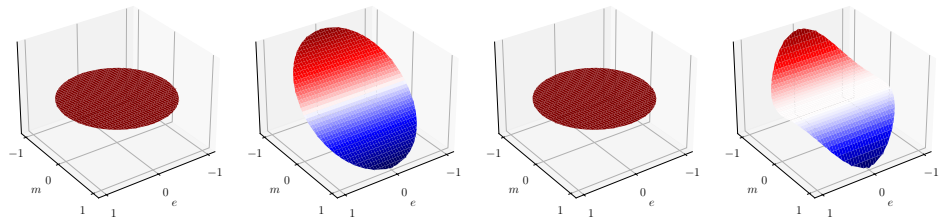


Figure F.1: Population loss of the model in Equation (F.15) with odd targets. (left) $g(\boldsymbol{z}_\star) = \mathrm{He}_1(\boldsymbol{z}_{\star,1}) + \mathrm{He}_1(\boldsymbol{z}_{\star,2})$, no positional encoding; (center-left) $g(\boldsymbol{z}_\star) = \mathrm{He}_1(\boldsymbol{z}_{\star,1}) + \mathrm{He}_1(\boldsymbol{z}_{\star,2})$, injected positional encoding; (center-right) $g(\boldsymbol{z}_\star) = \mathrm{He}_3(\boldsymbol{z}_{\star,1}) + \mathrm{He}_3(\boldsymbol{z}_{\star,2})$, no positional encoding; (right) $g(\boldsymbol{z}_\star) = \mathrm{He}_3(\boldsymbol{z}_{\star,1}) + \mathrm{He}_3(\boldsymbol{z}_{\star,2})$, injected positional encoding. $L = 2$, $\tilde{P} = 0$ in order to isolate the effect of the injection.

F.4 Sequence Information Exponent Beyond attention

In the main paper, we discussed the learning of a generic Sequence Single Index model with a parametrized model like Equation (6.1), with the goal of modelling the attention mechanism learning. The theory of Sequence Information Exponent goes beyond this, and can be used to understand the sample complexity of the learning of a generic sequence single-index model both as a target and as a trained model. In this Appendix, we focus on a particular choice of positional encoding that breaks the even symmetry of the model, allowing attention to weakly recover odd SIE targets. In particular, we consider a dynamical positional encoding of the form:

$$P_i = \frac{c_i}{\sqrt{d}}\boldsymbol{w} + \tilde{P}_i \qquad \text{with } \tilde{P}_i \in \mathbb{R}^{L\times d} \text{ a fixed vector.} \tag{F.14}$$

$\boldsymbol{c} \in \mathbb{R}^L$ is a fixed vector of coefficients. We call this special version of positional encoding *injected positional encoding*. The trained model becomes:

$$f_{\boldsymbol{w}}(X) = \Pi\left[\mathrm{softmax}\left(\left(X + \frac{\tilde{P}}{\sqrt{d}}\right)\boldsymbol{w}\boldsymbol{w}^\top\left(X + \frac{\tilde{P}}{\sqrt{d}}\right)^\top + \boldsymbol{c}\boldsymbol{c}^\top\right)\right], \tag{F.15}$$

and it has now a non-zero odd Hermite expansion. In Figure F.1 we show examples of population losses with odd targets, learned with the model in Equation (F.15). The injected positional encoding breaks the symmetry, as the population risk plots show.

F.5 Further analysis on the effect of sequence length

The aim of this Appendix is to clarify and give further examples on Theorem 6.2. Here we provide an intuitive explanation of the result, while the mathematical details can be found in Appendix F.3.

F.5.1 Anatomy of the gain

The gain counts *gradient steps* (Definition 6.1), while the analysis of Appendix F.3 is carried out on the gradient flow of Equation (6.11), which lives in continuous time. The learning rate is what converts one into the other, and it is precisely this conversion that decides whether the sequence length enters the gain once or twice. In this Appendix we write t_{tied} and t_{untied} for the weak recovery times of the gradient flow — called t_η and $t_{\eta,\text{untied}}$ in Appendix F.3 — and keep ν_η^{weak} for the number of gradient steps.

From continuous time to gradient steps Theorem F.2 relates the two, for a network whose gradient flow reaches weak recovery at time t. As long as the learning rate obeys

$$\gamma \lesssim \gamma^{\max} \sim \frac{1}{d\,t}, \tag{F.16}$$

weak recovery is reached after

$$\nu_\eta^{\text{weak}} \sim \frac{t}{\gamma} \tag{F.17}$$

gradient steps, whereas above $\gamma^{\max}$ the inhibitive term of Equation (6.11) dominates at initialization and the network never leaves the uninformative region. The key observation is that t appears in *both* equations: saturating Equation (F.16) gives $\nu_\eta^{\text{weak}} \sim d\,t^2$, so the weak recovery time of the flow is paid twice when the learning rate is itself scaled up with L, and only once when it is held fixed. This is the whole origin of the two regimes discussed below.

The two weak recovery times Close to initialization, both networks follow a flow of the same shape (Appendix F.3),

$$\frac{dm}{dt} \sim \alpha\, m^{\text{SIE}-1}, \tag{F.18}$$

and differ only through the drift constant α and the threshold θ that the overlap has to reach:

$$\begin{aligned} \alpha_{\text{tied}} &= C_{\text{SIE}} \times (\mathbf{1},\dots,\mathbf{1}), & \theta_{\text{tied}} &= \eta, \\ \alpha_{\text{untied}} &= \|C_{\text{SIE}}\|_{\text{op}}\,, & \theta_{\text{untied}} &= \eta\sqrt{L}. \end{aligned} \tag{F.19}$$

The drifts differ because of the structure of the two networks: the tied network builds up correlation faster than the untied one, since its single weight vector composes the L signals of the sequence coherently and therefore feels the target through its contraction along $(\mathbf{1},\dots,\mathbf{1})$; the untied network, having L independent weights, can only follow the leading direction of the tensor C_{SIE}, whose strength is measured by the operator norm. The thresholds differ for a purely geometric reason: weak recovery of the untied network is expressed through $m_{\text{untied}} = \|\boldsymbol{m}\|/\sqrt{L}$, so its overlap *vector* has to reach a norm $\sqrt{L}$ times larger than the scalar overlap of the tied network.

Integrating Equation (F.18) from the initial overlap $m(0) \sim 1/\sqrt{d}$ up to θ gives

$$t \sim \frac{1}{\alpha} \cdot \begin{cases} \theta & \text{SIE} = 1, \\ \ln d & \text{SIE} = 2, \\ d^{\frac{\text{SIE}}{2}-1} & \text{SIE} \geq 3, \end{cases} \tag{F.20}$$

and this is where the special role of SIE $= 1$ is born. When SIE $= 1$ the overlap grows linearly in time, so the weak recovery time is directly proportional to the threshold, and a threshold $\sqrt{L}$ times larger costs exactly $\sqrt{L}$ times more time. When SIE ≥ 2 the growth is instead exponential (SIE $= 2$) or explosive (SIE ≥ 3), and the weak recovery time is dominated by the escape from the small initial overlap $m(0)$: the threshold then enters only logarithmically, or through a bounded additive term, and its $\sqrt{L}$ is subleading. Dividing the two hitting times,

$$\frac{t_{\text{untied}}}{t_{\text{tied}}} \sim \frac{C_{\text{SIE}} \times (\mathbf{1}, \dots, \mathbf{1})}{\|C_{\text{SIE}}\|_{\text{op}}} \cdot \begin{cases} \sqrt{L} & \text{SIE} = 1, \\ 1 & \text{SIE} \geq 2. \end{cases} \tag{F.21}$$

Why the factor for SIE $= 1$ **is sometimes** $\sqrt{L}$ **and sometimes** L Writing Equation (F.17) for both networks, the gain splits into two factors,

$$\text{gain} = \frac{\nu^{\text{weak}}_{\eta,\text{untied}}}{\nu^{\text{weak}}_{\eta}} \sim \frac{t_{\text{untied}}}{t_{\text{tied}}} \cdot \frac{\gamma_{\text{tied}}}{\gamma_{\text{untied}}}, \tag{F.22}$$

and everything hinges on how the two learning rates are chosen. If the networks are trained with the same learning rate — or, more generally, with two rates whose ratio does not scale with L — the second factor is 1. If instead each network is given the largest learning rate it can afford, Equation (F.16) turns the second factor into a second copy of the first, because $\gamma^{\text{max}}_{\text{tied}}/\gamma^{\text{max}}_{\text{untied}} = t_{\text{untied}}/t_{\text{tied}}$. Hence

$$\text{gain} \sim \begin{cases} \dfrac{t_{\text{untied}}}{t_{\text{tied}}} & \text{equal learning rates,} \\ \left(\dfrac{t_{\text{untied}}}{t_{\text{tied}}}\right)^2 & \text{optimal learning rates.} \end{cases} \tag{F.23}$$

Substituting Equation (F.21) in the second case gives

$$\text{gain} \sim \left(\frac{C_{\text{SIE}} \times (\mathbf{1}, \dots, \mathbf{1})}{\|C_{\text{SIE}}\|_{\text{op}}}\right)^2 \cdot \begin{cases} L & \text{SIE} = 1, \\ 1 & \text{SIE} \geq 2, \end{cases} \tag{F.24}$$

which is exactly the statement of Theorem 6.2: the theorem is formulated at the optimal learning rate, as announced when the gain was defined. The extra factor attached to SIE $= 1$ is therefore *not* a constant of the model: it is $\sqrt{L}$ when the two networks share the same learning rate, and L when both rates are pushed to their maximum. It is one and the same effect — the $\sqrt{L}$ larger threshold of Equation (F.19) — counted once or twice, exactly like the drift ratio next to it.

For later use, combining Equations (F.16) and (F.20), the largest admissible learning rates scale with the sequence length as

$$\begin{aligned} \gamma^{\text{max}}_{\text{tied}} &\propto C_{\text{SIE}} \times (\mathbf{1}, \dots, \mathbf{1}), \\ \gamma^{\text{max}}_{\text{untied}} &\propto \|C_{\text{SIE}}\|_{\text{op}} \cdot \begin{cases} 1/\sqrt{L} & \text{SIE} = 1, \\ 1 & \text{SIE} \geq 2, \end{cases} \end{aligned} \tag{F.25}$$

up to factors that do not depend on L. Their ratio is again $t_{\text{untied}}/t_{\text{tied}}$, as it must be.

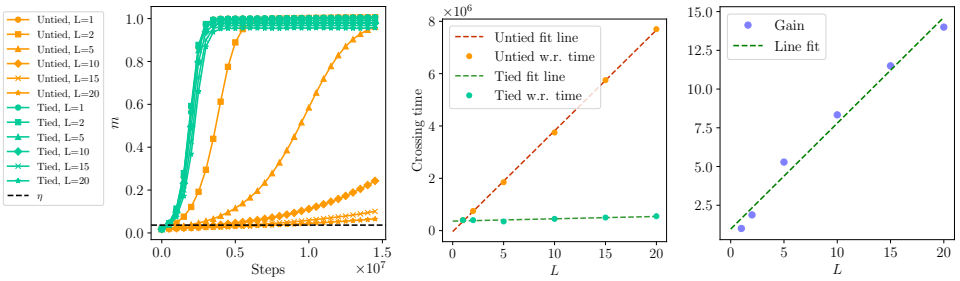


Figure F.2: The gain in the case of a target function with SIE $= 2$, where both cases use the same learning rate $\gamma_0 = 0.005$. (left) The evolution of the overlap, (middle) the weak recovery time and (right) the gain. The gain is proportional to L, as predicted. $d = 3000$, $\sigma = \text{ReLU}$

F.5.2 Example at non-optimal learning rate

In Figure 6.3 we presented the speed up in the case of a target function with SIE $= 2$, and optimal learning rate for both the tied and untied network. Here we want to show that our result predicts the gain well even when the learning rates are not scaled up optimally. In particular, consider the same setting as in Figure 6.3, but with the learning rates

$$\gamma_{\text{tied}} = \gamma_{\text{untied}} = \gamma_0 \qquad \text{independent of } L. \tag{F.26}$$

The target is $g(\boldsymbol{z}_\star) = \sum_{i=1}^{L} \text{He}_2(z_{\star,i})$, so that $C_2 = I_L$, $C_2 \times (\mathbf{1}, \mathbf{1}) = L$ and $\|C_2\|_{\text{op}} = 1$. Since SIE $= 2$ there is no threshold correction, and Equation (F.21) gives $t_{\text{untied}}/t_{\text{tied}} \sim L$. The two learning rates are equal, so we are in the first case of Equation (F.23) and the flow ratio is counted only once:

$$\text{gain} \sim \frac{t_{\text{untied}}}{t_{\text{tied}}} \sim L. \tag{F.27}$$

In Figure F.2 we show the result of the simulation, where we can see that the gain is indeed proportional to L, as predicted. It is worth stressing that this is the very same target function as in Figure 6.3, where the measured gain is instead L^2: the two experiments differ only in the choice of the learning rates, which decides whether the flow ratio is counted once or twice.

F.5.3 The upper bound on the learning rate scaling

In this subsection we want to show an example proving that not all scalings of the learning rate are allowed: if it grows too fast with the sequence length, the network will not be able to learn.

Let's take as example the SIE $= 1$ target function

$$g(\boldsymbol{z}_\star) = \frac{1}{\sqrt{L}} \sum_{i=1}^{L} z_{\star,i}, \tag{F.28}$$

with the corresponding leading Hermite tensor

$$C_1 = \begin{pmatrix} 1/\sqrt{L} & \dots & 1/\sqrt{L} \end{pmatrix} \in \mathbb{R}^L, \tag{F.29}$$

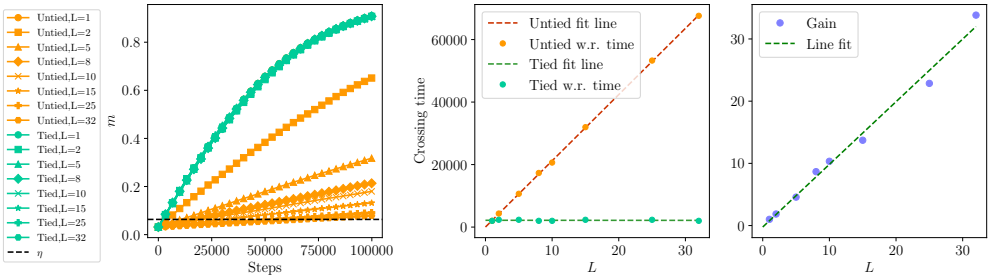


Figure F.3: The gain in the case of a target function with SIE $= 1$, where both cases use the same learning rate $\gamma_0 = 0.015$. (left) The evolution of the overlap, (middle) the weak recovery time and (right) the gain. The gain is proportional to L, as predicted. $d = 1000$, $\sigma = \mathrm{ReLU}$

for which $C_1 \times \mathbf{1} = \sqrt{L}$ and $\|C_1\|_{\mathrm{op}} = 1$. This time SIE $= 1$, so the threshold correction of Equation (F.21) is active and contributes a second $\sqrt{L}$:

$$\frac{t_{\mathrm{untied}}}{t_{\mathrm{tied}}} \sim \frac{C_1 \times \mathbf{1}}{\|C_1\|_{\mathrm{op}}} \cdot \sqrt{L} = \sqrt{L} \cdot \sqrt{L} = L. \tag{F.30}$$

We train both networks with one and the same L-independent learning rate

$$\gamma_{\mathrm{tied}} = \gamma_{\mathrm{untied}} = \gamma_0, \tag{F.31}$$

so that, once again, the flow ratio is counted only once:

$$\mathrm{gain} \sim \frac{t_{\mathrm{untied}}}{t_{\mathrm{tied}}} \sim L. \tag{F.32}$$

In Figure F.3 we show the result of the simulation, where we can see that the gain is indeed proportional to L, as predicted. Note that here the SIE $= 1$ correction shows up as a $\sqrt{L}$, and not as the L of Theorem 6.2, precisely because the learning rates are held fixed: had we instead scaled both of them up to their maximum, Equation (F.24) would have predicted gain $\sim (\sqrt{L})^2 \cdot L = L^2$. The two experiments of this Appendix therefore both measure a gain proportional to L, but for different reasons: for SIE $= 2$ the whole factor comes from the drift ratio, while for SIE $= 1$ half of it comes from the drift ratio and half from the larger threshold of the untied network.

We can now ask what happens if we push the learning rate scaling of the untied network beyond the limit of Theorem 6.2. If we keep $\gamma_{\mathrm{tied}} = \gamma_0$ and set

$$\gamma_{\mathrm{untied}} = \gamma_0 \cdot L, \tag{F.33}$$

the ratio between the learning rates becomes now $\gamma_{\mathrm{tied}}/\gamma_{\mathrm{untied}} = 1/L$, and Equation (F.22) would give

$$\mathrm{gain} \sim L \cdot \frac{1}{L} = 1. \tag{F.34}$$

Apparently, the untied network performance is matching the one of the tied network. The prediction is however void, because Equation (F.17) only holds below the threshold of Equation (F.16):

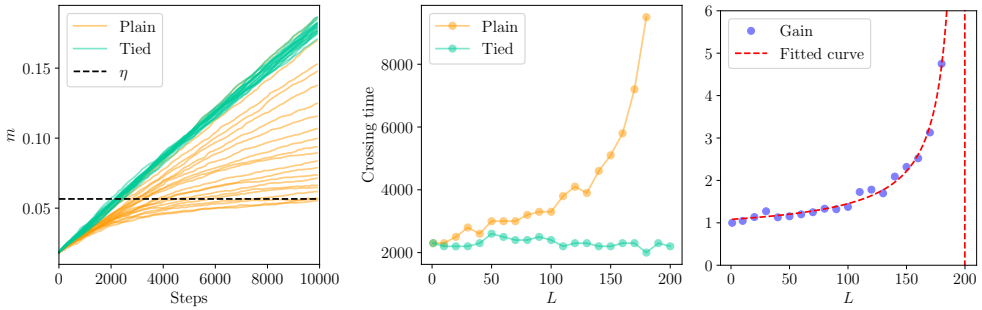


Figure F.4: The gain in the case of a target function with SIE $= 1$, where the untied network is trained with $\gamma_{\text{untied}} = \gamma_0 L$, beyond the bound of Equation (F.16), while the tied one keeps $\gamma_{\text{tied}} = \gamma_0 = 0.05$. (left) The evolution of the overlap, (middle) the weak recovery time and (right) the gain. The gain diverges, as predicted. $d = 3000$, $\sigma = \text{ReLU}$

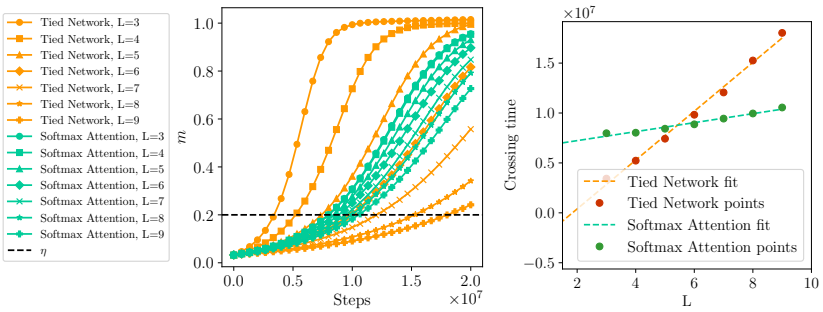


Figure F.5: Comparison between the learning of the tied network with squared activation (orange), namely linear attention, and the single-layer softmax attention (green). (left) The evolution of the overlap; each symbol is a different value of L. (right) the weak recovery time. The two networks learn at the same rate, but with different constants. $g(\boldsymbol{z}_\star) = \sum_{i=1}^{L} \text{He}_2(\boldsymbol{z}_{\star,i})$, $d = 1000$, $\sigma = \text{ReLU}$

for this target the largest admissible rate of the untied network *decreases* with the sequence length as $1/\sqrt{L}$, by Equation (F.25), so a learning rate growing as $\gamma_0 L$ overshoots it by a factor $L^{3/2}$. Figure F.4 shows the result of the simulation of such a case: the learning rate of the untied network is too high, the untied network stops reaching weak recovery as L grows, and the measured gain diverges instead of settling to a constant. This plot proves our bounds on the learning rate scaling are indeed correct.

F.5.4 Equivalence between Single-Layer attention and Tied network

In the main paper we proved that the tied network model is a generalization of the single-layer attention model. In particular, if the activation function is $\sigma(x) = x^2$, the tied network is equivalent to the single-layer *linear* attention. We also claimed that adding a non-linearity to the attention can only improve the performance of the network, although not affecting the scaling with the sequence length. In Figure F.5 we show the comparison between the learning of the

tied network with squared activation (orange), namely linear attention, and the single-layer softmax attention(green). The two networks learn at the same linear rate, but with different growth constants. Softmax attention performs better than the tied network with squared activation for sufficiently large sequence lengths.

F.5.5 Pathological cases

Theorem 6.2 shows that there could be cases where the gain from learning is 0, meaning that the tied network cannot learn anything, while the untied one possibly can. In particular the degenerate condition happens when

$$C_{\text{SIE}} \times (\mathbf{1}, \dots, \mathbf{1}) = 0, \tag{F.35}$$

where Theorem 6.2 guarantees that the gain is 0. Examples of such pathological cases are:

- SIE $= 1$: a possible pathological target could be

  $$g(\boldsymbol{z}_\star) = z_{\star,1} - z_{\star,2}.$$

  In this case the leading term in Hermite expansion is

  $$C_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \text{and consequently } C_1 \times \mathbf{1} = 0.$$

- SIE $= 2$: analogously, a possible pathological target could be

  $$g(\boldsymbol{z}_\star) = z_{\star,1}^2 - z_{\star,2}^2.$$

  In this case the leading term in Hermite expansion is

  $$C_2 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and consequently } C_2 \times \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = 0.$$

In Figure F.6 we show the pathological case for SIE $= 1$ and SIE $= 2$. The untied network is not able to learn the target function because of the symmetry: the tied network is by design symmetric, while the target is constructed to be as antisymmetric as possible in the sequence length. The untied network instead is able to learn the target function, because the weights are not constrained to be equal, thus the symmetry is broken by the initialization.

The degeneracy of the tied network could be solved by weighting randomly the neurons. In that case, almost surely we have

$$C_{\text{SIE}} \times (\mathbf{1}, \dots, \mathbf{1}) \neq 0. \tag{F.36}$$

Although this is a possible solution, we leave the study of the effect of random weights for symmetry breaking for future work.

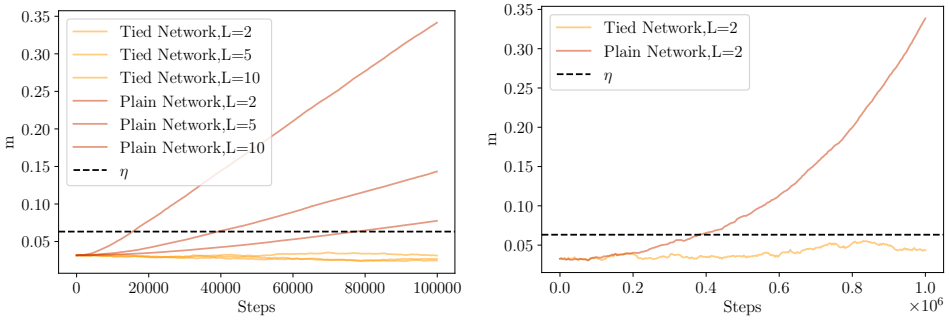


Figure F.6: Pathological cases for SIE $= 1$ (left) and SIE $= 2$ (right). The untied network is able to learn the target function, while the tied one cannot.

F.6 Further discussion on the positional-semantic transition

F.6.1 SIE of model (6.18)

Definition 6.2 of the sequence information exponent does not apply directly to the model (6.18) because the model has a non-scalar output. However, we can still extend the concept of SIE to this case by looking at the update rule of sufficient statistics around initialization.

The population loss function is given by

$$\begin{aligned} R(e,m) &= \mathbb{E}_{\boldsymbol{z},\boldsymbol{z}_\star\sim\mathcal{P}(e,m)}\left[\left\|(1-\omega)\,\mathrm{softmax}\left(\boldsymbol{z}_\star\boldsymbol{z}_\star^\top\right)+\omega\,\mathrm{softmax}\begin{bmatrix}\begin{pmatrix}a^2 & -a^2\\ -a^2 & a^2\end{pmatrix}\end{bmatrix}\right.\right. \\ &\qquad\left.\left.-\,\mathrm{softmax}\left(\boldsymbol{z}\boldsymbol{z}^\top\right)\right\|_F^2\right] \\ &= \mathbb{E}_{\boldsymbol{z},\boldsymbol{z}_\star\sim\mathcal{P}(e,m)}\left[\mathcal{L}(\boldsymbol{z},\boldsymbol{z}_\star)\right] \end{aligned} \tag{F.37}$$

with

$$\mathcal{P}(e,m)\equiv\mathcal{N}\left(\begin{pmatrix}0\\0\\e\\-e\end{pmatrix},\begin{pmatrix}1&0&m&0\\0&1&0&m\\m&0&1&0\\0&m&0&1\end{pmatrix}\right). \tag{F.38}$$

Using parity arguments, we can easily show that the gradient of the population loss is 0 at the initialization point $(e,m)=(0,0)$. Let $p(\boldsymbol{z}_\star,\boldsymbol{z};e,m)$ be the probability density function associated with the distribution $\mathcal{P}(e,m)$. Then the gradient at initialization is

$$\nabla_{(e,m)}R(e,m)\Big|_{e=m=0}=\mathbb{E}_{\boldsymbol{z},\boldsymbol{z}_\star\sim\mathcal{P}(e,m)}\left[\frac{\nabla_{(e,m)}p(\boldsymbol{z}_\star,\boldsymbol{z};e,m)}{p(\boldsymbol{z}_\star,\boldsymbol{z};0,0)}\mathcal{L}(\boldsymbol{z},\boldsymbol{z}_\star)\right] \tag{F.39}$$

$\mathcal{L}(\boldsymbol{z}_\star, \boldsymbol{z})$ is an even function of $(\boldsymbol{z}_\star, \boldsymbol{z})$, while the gradient of the probability density function is an odd function of $(\boldsymbol{z}_\star, \boldsymbol{z})$. Therefore, the product is an odd function of $(\boldsymbol{z}_\star, \boldsymbol{z})$ and the expectation is 0.

$$\nabla_{(e,m)} R(e,m)\Big|_{e=m=0} = (0,0). \tag{F.40}$$

We can use the same computation technique to compute the Hessian of the population loss at initialization. This time the parity arguments does not apply, and we can show numerically that the Hessian is non-null

$$\frac{\partial^2 R}{\partial e^2}\bigg|_{e=m=0}, \frac{\partial^2 R}{\partial m^2}\bigg|_{e=m=0}, \frac{\partial^2 R}{\partial e \partial m}\bigg|_{e=m=0} \neq 0. \tag{F.41}$$

The information exponent We can also compute the *information exponent* by looking at what rate m grows around initialization. We use *spherical gradient descent* because we assumed the norm of the vector $\boldsymbol{w}$ to be constant (as Ben Arous also does in his paper on Information Exponent):

The update rule of $\boldsymbol{w}$ is

$$\boldsymbol{w}^{\nu+1} = \frac{\boldsymbol{w}^\nu - \gamma \nabla_{\boldsymbol{w}} \mathcal{L}}{\|\boldsymbol{w}^\nu - \gamma \nabla_{\boldsymbol{w}} \mathcal{L}\|_2} \sqrt{d},$$

multiplying both sides by $\boldsymbol{w}_\star/d$ we get

$$m^{\nu+1} = \frac{m^\nu - \gamma \frac{\boldsymbol{w}_\star \cdot \nabla_{\boldsymbol{w}} \mathcal{L}}{d}}{\|\boldsymbol{w}^\nu - \gamma \nabla_{\boldsymbol{w}} \mathcal{L}\|_2} \sqrt{d}.$$

We can assume to be in the *gradient flow regime*, where the learning rate $\gamma \ll 1$

$$\|\boldsymbol{w} - \gamma \nabla_{\boldsymbol{w}} \mathcal{L}\|_2 = \sqrt{\|w\|^2 - 2\gamma \boldsymbol{w} \cdot \nabla_w \mathcal{L} + \gamma^2 \|\nabla_w \mathcal{L}\|^2} \approx \sqrt{d}\sqrt{1 - 2\gamma \frac{\boldsymbol{w} \cdot \nabla_w \mathcal{L}}{d}},$$

that finally lead us to

$$m^{\nu+1} = \left(m^\nu - \gamma \frac{\boldsymbol{w}_\star \cdot \nabla_{\boldsymbol{w}} \mathcal{L}}{d}\right)\left(1 + \gamma \frac{\boldsymbol{w} \cdot \nabla_w \mathcal{L}}{d}\right) \approx m^\nu - \gamma \frac{\boldsymbol{w}_\star \cdot \nabla_{\boldsymbol{w}} \mathcal{L}}{d} + m^\nu \gamma \frac{\boldsymbol{w} \cdot \nabla_w \mathcal{L}}{d}, \tag{F.42}$$

where in the last step we used again the small learning rate limit.

Now we can use the chain rule for getting the gradient in terms of the derivatives we already have

$$\nabla_w \mathcal{L} = \frac{\partial \mathcal{L}}{\partial m} \cdot \nabla_{\boldsymbol{w}} m + \frac{\partial \mathcal{L}}{\partial e} \cdot \nabla_{\boldsymbol{w}} e = \frac{\partial \mathcal{L}}{\partial m} \cdot \frac{\boldsymbol{w}_\star}{d} + \frac{\partial \mathcal{L}}{\partial e} \cdot \frac{\boldsymbol{p}}{\sqrt{d}}$$

We can rearrange the equation above as

$$\begin{aligned}
\frac{m^{\nu+1} - m^\nu}{\gamma/d} &= -\boldsymbol{w}_\star \cdot \left(\frac{\partial \mathcal{L}}{\partial m} \cdot \frac{\boldsymbol{w}_\star}{d} + \frac{\partial \mathcal{L}}{\partial e} \cdot \frac{\boldsymbol{p}}{\sqrt{d}}\right) + \boldsymbol{w} \cdot \left(\frac{\partial \mathcal{L}}{\partial m} \cdot \frac{\boldsymbol{w}_\star}{d} + \frac{\partial \mathcal{L}}{\partial e} \cdot \frac{\boldsymbol{p}}{\sqrt{d}}\right) \\
&= -\frac{\partial \mathcal{L}}{\partial m} + m \frac{\partial \mathcal{L}}{\partial m} + e \frac{\partial \mathcal{L}}{\partial e}
\end{aligned}$$

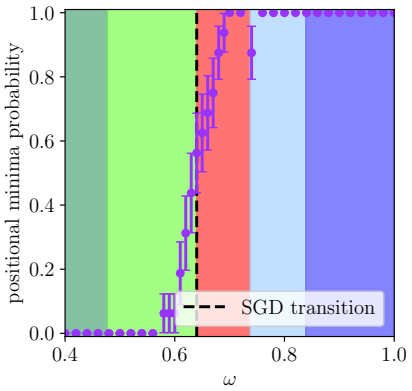


Figure F.7: Reproduction of Figure 6.5 for $d = 100$. The transition is smoother than in the $d = 1000$. Averaged over 25 runs.

where we used the fact $\boldsymbol{p} \cdot \boldsymbol{w}_\star \approx 0$ in high dimension (as already assumed above), and $\|\boldsymbol{w}_\star\| = d$.

We are interested in what is happening at initialization, therefore all the derivatives should be evaluated around $(e, m) = 0$. We already know that $\nabla_{e,m}\mathcal{L}\big|_{e=m=0} = 0$, so we expand the at the next order in m

$$\frac{m^{\nu+1} - m^\nu}{\gamma/d} = -m \left.\frac{\partial^2 \mathcal{L}}{\partial m^2}\right|_{m,e=0} - e \left.\frac{\partial^2 \mathcal{L}}{\partial m \partial e}\right|_{m,e=0} .$$

We can repeat the same derivation for finding the analogous equation for e:

$$\frac{e^{\nu+1} - e^\nu}{\gamma/d} = -e \left.\frac{\partial^2 \mathcal{L}}{\partial e^2}\right|_{m,e=0} - m \left.\frac{\partial^2 \mathcal{L}}{\partial m \partial e}\right|_{m,e=0} .$$

Since the Hessian of the loss has always a negative eigenvalue, both these equations need $\nu = O(d \log d)$ steps of population gradient descent at constant learning rate γ to escape a neighborhood of initialization, leading to SIE $= 2$. This step count is not the sample complexity of online SGD: controlling the gradient noise forces $\gamma = O(1/\log d)$, and the same estimate $\nu \asymp d/\gamma \cdot \log d$ then returns $\mathcal{O}(d \log^2 d)$ — the extra logarithmic factor that Theorem 6.1 suppresses when it quotes the sharp $\mathcal{O}(d \log d)$.

F.6.2 Numerical experiments on the transition

In this Appendix we would like to provide more details on the phase diagram of Figure 6.4.

In Figure F.7 we reproduce Figure 6.5 for $d = 100$, clarifying the *finite size* effects we claimed in the main text. Smaller values of d lead to a smoother transition, since the gradient flow assumption is less valid. In the limit $d \to \infty$ we expect to see a step function.

The yellow region in the phase diagram of Figure 6.4 is predicted to have a unique global semantic minimum, while SGD is aligned with the positional direction at initialization. The resulting

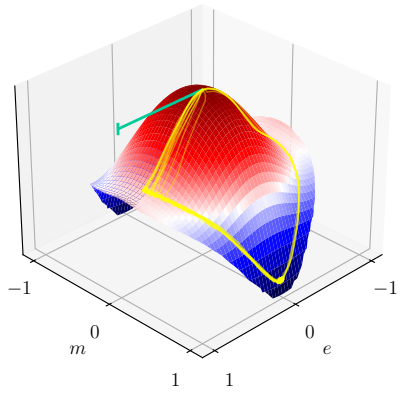


Figure F.8: The loss surface in the yellow region of the phase diagram. The SGD trajectory is shown in yellow. The training moves towards the direction orthogonal to the one of global minima, ultimately slowing down the convergence.

trajectories are shown in Figure F.8: the dynamics moves towards the minima direction, get stuck in the local flat (but not critical) region around $(e, m) = (1, 0)$, and then it moves towards the global minima. In this region the convergence is very slow. The turning point of the dynamics is a bit misplaced because of the numerical errors given by the loss integration; details on this in Appendix F.7.

F.6.3 An alternative model without phase transition

In order to highlight the peculiarity of the model (6.18), we present an example of a model that does not exhibit a phase transition between the positional and semantic regimes. Let's assume that our target is a softmax attention matrix *with* positional encoding

$$y(X) = \text{softmax}\left[(X + {}^{P_\star}/\sqrt{d})\boldsymbol{w}_\star\boldsymbol{w}_\star^\top(X + {}^{P_\star}/\sqrt{d})^\top\right]$$

where

$$P_\star = \begin{pmatrix} +\boldsymbol{p}_\star \\ -\boldsymbol{p}_\star \end{pmatrix}, \qquad \boldsymbol{w}_\star = \sqrt{1-\omega^2}\boldsymbol{w}_s + \omega\boldsymbol{p}_\star\sqrt{d} \quad \text{with} \quad \boldsymbol{w}_s \perp \boldsymbol{p}_\star.$$

ω plays the same role as in the model (6.18), and we can set $\omega = 0$ to get the semantic model, or $\omega = 1$ to get the positional model. The difference is that in this case the positional encoding is added to the input of the softmax function, while in the previous model it was added to the output. This change has a significant impact on the behavior of the model.

The global minima of the population loss function does not transition from a positional to a semantic regime, but rather it smoothly move from the semantic to the positional regime as ω increases. We leave the details and numerical experiments for future work.

F.7 Numerical experiments details

All the codes used to run the experiments are available at this GitHub repository[1]; where details for reproducing figures are not available in the paper, we provide the code to reproduce them in the repository.

The experiments run on a Mac Studio M2 Ultra, within at most few hours for the largest ones. The code is written in Python, using the libraries `numpy`, `scipy`, `torch` and `matplotlib`. `hydra` is used to manage the configuration files.

F.7.1 Details on the integration method of squared loss

All the plot of population loss we showed are a numerical integration of the loss function. As showed in Section 6.1, the population loss is given by a multivariate Gaussian integral of $2L$ dimensions, where the mean and the covariance are determined by the sufficient statistics. The integral can't be computed analytically, so we use a custom numerical procedure based on the Gauss-Hermite quadrature.

Let $f\colon \mathbb{R}^{2L} \to \mathbb{R}$ be a function of $2L$ variables to be integrated, and let $\mu \in \mathbb{R}^{2L}$ and $\Sigma \in \mathbb{R}^{2L\times 2L}$ be the mean and covariance of the multivariate Gaussian distribution. The integral we want to compute is

$$I = \int_{\mathbb{R}^{2L}} f(x) \frac{1}{(2\pi)^L \sqrt{\det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right) dx \tag{F.43}$$

The numerical procedure is as follows:

(i) Compute the 1D Gauss-Hermite nodes and weights for N_{int} points

$$\{x_i\}_{i=1}^{N_{\text{int}}} \text{ and } \{w_i\}_{i=1}^{N_{\text{int}}}$$

where x_i are the nodes given by the roots of the Hermite polynomial $H_{N_{\text{int}}}(x)$ and w_i are the corresponding weights, computed as

$$w_i = \frac{2^{N_{\text{int}}+1} N_{\text{int}}! \sqrt{\pi}}{H'_{N_{\text{int}}}(x_i)^2}$$

(ii) Compute the $2L$ dimensional nodes and weights by taking the Kronecker product of the 1D nodes and weights

$$\{X_i\}_{i=1}^{N_{\text{int}}^{2L}} = \bigotimes_{l=1}^{2L} \{x_i\}_{i=1}^{N_{\text{int}}} \text{ and } \{W_i\}_{i=1}^{N_{\text{int}}^{2L}} = \bigotimes_{l=1}^{2L} \{w_i\}_{i=1}^{N_{\text{int}}}$$

[1] `https://github.com/IdePHICS/Sequence-Single-Index`

(iii) Let T be the Cholesky decomposition of Σ, i.e. $\Sigma = T^\top T$. We can then change the variable to $y = T^{-1}(x - \mu)$ and compute the integral as

$$\begin{aligned} I &= \int_{\mathbb{R}^{2L}} f(x) \frac{1}{(2\pi)^L \sqrt{\det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right) dx \\ &= \int_{\mathbb{R}^{2L}} f(Ty+\mu) \frac{1}{(2\pi)^L} \exp\left(-\frac{1}{2} y^\top y\right) dy \end{aligned}$$

(iv) The integral can then be approximated, as

$$I \approx \sum_{i=1}^{N_{\text{int}}^{2L}} \left(\prod_{l=1}^{2L} W_{i,l} \right) f(TX_i + \mu)$$

The precision of the integral is obviously regulated by the number of points N_{int} we use. In our experiments, we used $N_{\text{int}} = 17$, while for the phase diagram in Figure 6.4 we used $N_{\text{int}} = 19$. Some effects of this integration error are visible in Figure F.8.

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". In: *Proceedings of Thirty Fifth Conference on Learning Theory*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 4782–4887.

[AAM23] Emmanuel Abbe, Enric Boix Adserà, and Theodor Misiakiewicz. "SGD learning on neural networks: leap complexity and saddle-to-saddle dynamics". In: *Proceedings of Thirty Sixth Conference on Learning Theory*. Ed. by Gergely Neu and Lorenzo Rosasco. Vol. 195. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 2552–2623. URL: https://proceedings.mlr.press/v195/abbe23a.html.

[Abb+21] Emmanuel Abbe et al. "The staircase property: How hierarchical structure can guide deep learning". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 26989–27002.

[ABP22] Alexander Atanasov, Blake Bordelon, and Cengiz Pehlevan. "Neural Networks as Kernel Learners: The Silent Alignment Effect". In: *International Conference on Learning Representations*. 2022.

[ACH18] Sanjeev Arora, Nadav Cohen, and Elad Hazan. "On the Optimization of Deep Networks: Implicit Acceleration by Overparameterization". In: *Proceedings of the 35th International Conference on Machine Learning (ICML)*. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 244–253.

[Ana+17] Anima Anandkumar et al. "Homotopy Analysis for Tensor PCA". In: *Proceedings of the 2017 Conference on Learning Theory*. Ed. by Satyen Kale and Ohad Shamir. Vol. 65. Proceedings of Machine Learning Research. PMLR, July 2017, pp. 79–104. URL: https://proceedings.mlr.press/v65/anandkumar17a.html.

[Arn+23] Luca Arnaboldi et al. "From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layers networks". In: *Proceedings of Thirty Sixth Conference on Learning Theory*. Ed. by Gergely Neu and Lorenzo Rosasco. Vol. 195. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 1199–1227. URL: https://proceedings.mlr.press/v195/arnaboldi23a.html.

[Arn+24a] Luca Arnaboldi et al. "Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD". In: *arXiv preprint arXiv:2305.18502* (2024).

[Arn+24b] Luca Arnaboldi et al. "Online Learning and Information Exponents: The Importance of Batch size & Time/Complexity Tradeoffs". In: *Proceedings of the 41st International Conference on Machine Learning*. Vol. 235. Proceedings of Machine Learning Research. PMLR, July 2024, pp. 1730–1762.

[Arn+24c] Luca Arnaboldi et al. "Repetita iuvant: Data repetition allows sgd to learn high-dimensional multi-index functions". In: *arXiv preprint arXiv:2405.15459* (2024).

[Aub+20] Benjamin Aubin et al. "Exact asymptotics for phase retrieval and compressed sensing with random generative priors". In: *Proceedings of The First Mathematical and Scientific Machine Learning Conference*. Vol. 107. Proceedings of Machine Learning Research. PMLR, July 2020, pp. 55–73.

[Ba+22] Jimmy Ba et al. "High-dimensional Asymptotics of Feature Learning: How One Gradient Step Improves the Representation". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 37932–37946.

[Ba+23] Jimmy Ba et al. "Learning in the Presence of Low-dimensional Structure: A Spiked Random Matrix Perspective". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Vol. 36. Curran Associates, Inc., 2023, pp. 17420–17449. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/38a1671ab0747b6ffe4d1c6ef117a3a9-Paper-Conference.pdf.

[Bar+19] Jean Barbier et al. "Optimal errors and phase transitions in high-dimensional generalized linear models". In: *Proceedings of the National Academy of Sciences* 116.12 (2019), pp. 5451–5460. DOI: 10.1073/pnas.1802705116[2]. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1802705116. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1802705116.

[BB07] Léon Bottou and Olivier Bousquet. "The Tradeoffs of Large Scale Learning". In: *Advances in Neural Information Processing Systems*. Ed. by J. Platt et al. Vol. 20. Curran Associates, Inc., 2007.

[BBP23] Alberto Bietti, Joan Bruna, and Loucas Pillaud-Vivien. "On Learning Gaussian Multi-index Models with Gradient Flow". In: *arXiv preprint arXiv:2310.19793* (2023).

[BC03] Léon Bottou and Yann Cun. "Large Scale Online Learning". In: *Advances in Neural Information Processing Systems*. Ed. by S. Thrun, L. Saul, and B. Schölkopf. Vol. 16. MIT Press, 2003. URL: https://proceedings.neurips.cc/paper_files/paper/2003/file/9fb7b048c96d44a0337f049e0a61ff06-Paper.pdf.

[BC22] Francis Bach and Lénaic Chizat. "Gradient Descent on Infinitely Wide Neural Networks: Global Convergence and Generalization". In: *International Congress of Mathematicians*. 2022.

[BD20] Björn Barz and Joachim Denzler. "Do we train on test data? Purging cifar of near-duplicates". In: *Journal of Imaging* 6.6 (2020), p. 41.

[2] https://doi.org/10.1073/pnas.1802705116

[BGJ21] Gerard Ben Arous, Reza Gheissari, and Aukosh Jagannath. "Online stochastic gradient descent on non-convex losses from high-dimensional inference". In: *The Journal of Machine Learning Research* 22.1 (2021), pp. 4788–4838.

[BGJ22] Gerard Ben Arous, Reza Gheissari, and Aukosh Jagannath. "High-dimensional limit theorems for sgd: Effective dynamics and critical scaling". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 25349–25362.

[Bie+22] Alberto Bietti et al. "Learning single-index models with shallow neural networks". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 9768–9783.

[BLM13] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford, New York: Oxford University Press, Feb. 2013. ISBN: 9780199535255.

[BMZ24] Raphaël Berthier, Andrea Montanari, and Kangjie Zhou. "Learning time-scales in two-layers neural networks". In: *Foundations of Computational Mathematics* (2024), pp. 1–84.

[Bor+24] Blake Bordelon et al. "Depthwise Hyperparameter Transfer in Residual Networks: Dynamics and Scaling Limit". In: *International Conference on Learning Representations (ICLR)*. arXiv:2309.16620. 2024.

[BPF22] Etienne Boursier, Loucas Pillaud-Vivien, and Nicolas Flammarion. "Gradient flow dynamics of shallow relu networks for square loss and orthogonal inputs". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 20105–20118.

[Bro+20] Tom Brown et al. *Language models are few-shot learners*. 2020.

[BRW96] Michael Biehl, Peter Riegler, and Christian Wöhler. "Transient dynamics of on-line learning in two-layered neural networks". In: *Journal of Physics A: Mathematical and General* 29.16 (Aug. 1996), pp. 4769–4780. DOI: 10.1088/0305-4470/29/16/005[3].

[BS95] M Biehl and H Schwarze. "Learning by on-line gradient descent". In: *Journal of Physics A: Mathematical and General* 28.3 (Feb. 1995), pp. 643–656. DOI: 10.1088/0305-4470/28/3/018[4].

[Cag+24] Francesco Cagnetta et al. "How deep neural networks learn compositional data: The random hierarchy model". In: *Physical Review X* 14.3 (2024), p. 031001.

[Can+13] Emmanuel J. Candès et al. "Phase Retrieval via Matrix Completion". In: *SIAM Journal on Imaging Sciences* 6.1 (2013), pp. 199–225. DOI: 10.1137/110848074[5].

[CB18] Lénaic Chizat and Francis Bach. "On the Global Convergence of Gradient Descent for Over-parameterized Models using Optimal Transport". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018.

[3] https://doi.org/10.1088/0305-4470/29/16/005

[4] https://doi.org/10.1088/0305-4470/28/3/018

[5] https://doi.org/10.1137/110848074

[CC95] M Copelli and N Caticha. "On-line learning in the committee machine". In: *Journal of Physics A: Mathematical and General* 28.6 (Mar. 1995), pp. 1615–1625. DOI: 10.1088/0305-4470/28/6/016[6].

[CCM21] Michael Celentano, Chen Cheng, and Andrea Montanari. "The high-dimensional asymptotics of first order methods with random data". In: *arXiv preprint arXiv:2112.07572* (2021).

[Che+19] Yuxin Chen et al. "Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval". In: *Mathematical Programming* 176.1 (July 2019), pp. 5–37. ISSN: 1436-4646. DOI: 10.1007/s10107-019-01363-6[7].

[Che17] A. Chertock. "A Practical Guide to Deterministic Particle Methods". en. In: *Handbook of Numerical Methods for Hyperbolic Problems: Applied and Modern Issues*. Ed. by Rémi Abgrall and Chi-Wang Shu. Vol. 18. Handbook of Numerical Analysis. Elsevier, Jan. 2017. Chap. 7, pp. 177–202. DOI: 10.1016/bs.hna.2016.11.004[8].

[Chu54] K. L. Chung. "On a Stochastic Approximation Method". In: *The Annals of Mathematical Statistics* 25.3 (1954), pp. 463–483. DOI: 10.1214/aoms/1177728716[9].

[CLS15] Emmanuel J. Candès, Xiaodong Li, and Mahdi Soltanolkotabi. "Phase Retrieval via Wirtinger Flow: Theory and Algorithms". In: *IEEE Transactions on Information Theory* 61.4 (2015), pp. 1985–2007. DOI: 10.1109/TIT.2015.2399924[10].

[CM20] Sitan Chen and Raghu Meka. "Learning Polynomials in Few Relevant Dimensions". In: *Proceedings of Thirty Third Conference on Learning Theory*. Vol. 125. Proceedings of Machine Learning Research. PMLR, July 2020, pp. 1161–1227.

[Col+24] Elizabeth Collins-Woodfin et al. "Hitting the High-dimensional notes: an ODE for SGD learning dynamics on GLMs and multi-index models". In: *Information and Inference: A Journal of the IMA* (2024).

[CPT21] Kabir Aladin Chandrasekher, Ashwin Pananjady, and Christos Thrampoulidis. "Sharp global convergence guarantees for iterative nonconvex optimization: A Gaussian process perspective". In: *arXiv preprint arXiv:2109.09859* (2021).

[CRB24] Otávio Cistolo Citton, Frederieke Richert, and Michael Biehl. "On-line learning dynamics in layered neural networks with arbitrary activation functions". In: *ESANN 2024 proceedings: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*. 2024, pp. 437–442.

[CS00] A. C. C. Coolen and D. Saad. "Dynamics of learning with restricted training sets". In: *Phys. Rev. E* 62 (4 Oct. 2000), pp. 5444–5487. DOI: 10.1103/PhysRevE.62.5444[11]. URL: https://link.aps.org/doi/10.1103/PhysRevE.62.5444.

[6] https://doi.org/10.1088/0305-4470/28/6/016
[7] https://doi.org/10.1007/s10107-019-01363-6
[8] https://doi.org/10.1016/bs.hna.2016.11.004
[9] https://doi.org/10.1214/aoms/1177728716
[10] https://doi.org/10.1109/TIT.2015.2399924
[11] https://doi.org/10.1103/PhysRevE.62.5444

[CSV13] Emmanuel J. Candès, Thomas Strohmer, and Vladislav Voroninski. "PhaseLift: Exact and Stable Signal Recovery from Magnitude Measurements via Convex Programming". In: *Communications on Pure and Applied Mathematics* 66.8 (2013), pp. 1241–1274. DOI: https://doi.org/10.1002/cpa.21432[12].

[CSX00] A. C. C. Coolen, D. Saad, and Yuan-Sheng Xiong. "On-line learning from restricted training sets in multilayer neural networks". In: *Europhysics Letters* 51.6 (Sept. 2000), p. 691. DOI: 10.1209/epl/i2000-00394-5[13]. URL: https://dx.doi.org/10.1209/epl/i2000-00394-5.

[Cui+24a] Hugo Cui et al. "A Phase Transition between Positional and Semantic Learning in a Solvable Model of Dot-Product Attention". In: *Advances in Neural Information Processing Systems*. Vol. 37. Curran Associates, Inc., 2024, pp. 36342–36389.

[Cui+24b] Hugo Cui et al. "Asymptotics of feature learning in two-layer networks after one gradient-step". In: *arXiv preprint arXiv:2402.04980* (2024).

[Cui25] Hugo Cui. "High-dimensional learning of narrow neural networks". In: *Journal of Statistical Mechanics: Theory and Experiment* 2025.2 (2025), p. 023402.

[Dam+23] Alex Damian et al. "Smoothing the Landscape Boosts the Signal for SGD: Optimal Sample Complexity for Learning Single Index Models". In: *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*. Ed. by Alice Oh et al. 2023. URL: http://papers.nips.cc/paper%5C_files/paper/2023/hash/02763667a5761ff92bb15d8751bcd223-Abstract-Conference.html.

[Dam+24] Alex Damian et al. "Computational-Statistical Gaps in Gaussian Single-Index Models". In: *The Thirty Seventh Annual Conference on Learning Theory*. PMLR. 2024, pp. 1262–1262.

[Dan+24a] Yatin Dandi et al. "How two-layer neural networks learn, one (giant) step at a time". In: *Journal of Machine Learning Research* 25.349 (2024), pp. 1–65.

[Dan+24b] Yatin Dandi et al. "The Benefits of Reusing Batches for Gradient Descent in Two-Layer Networks: Breaking the Curse of Information and Leap Exponents". In: *Proceedings of the 41st International Conference on Machine Learning*. Ed. by Ruslan Salakhutdinov et al. Vol. 235. Proceedings of Machine Learning Research. PMLR, July 2024, pp. 9991–10016. URL: https://proceedings.mlr.press/v235/dandi24a.html.

[DDB20] Aymeric Dieuleveut, Alain Durmus, and Francis Bach. "Bridging the gap between constant step size stochastic gradient descent and Markov chains". In: *The Annals of Statistics* 48.3 (2020), pp. 1348–1382. DOI: 10.1214/19-AOS1850[14].

[DDP20] Damek Davis, Dmitriy Drusvyatskiy, and Courtney Paquette. "The nonsmooth landscape of phase retrieval". In: *IMA Journal of Numerical Analysis* 40.4 (Jan. 2020), pp. 2652–2695. ISSN: 0272-4979. DOI: 10.1093/imanum/drz031[15].

[12] https://doi.org/https://doi.org/10.1002/cpa.21432
[13] https://doi.org/10.1209/epl/i2000-00394-5
[14] https://doi.org/10.1214/19-AOS1850
[15] https://doi.org/10.1093/imanum/drz031

[Def+25] Leonardo Defilippis et al. "Optimal Spectral Transitions in High-Dimensional Multi-Index Models". In: *arXiv preprint arXiv:2502.02545* (2025).

[Dev+18] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *CoRR* abs/1810.04805 (2018). arXiv: 1810.04805[16].

[DFB17] Aymeric Dieuleveut, Nicolas Flammarion, and Francis Bach. "Harder, Better, Faster, Stronger Convergence Rates for Least-Squares Regression". In: *Journal of Machine Learning Research* 18.101 (2017), pp. 1–51.

[DH18] Rishabh Dudeja and Daniel Hsu. "Learning Single-Index Models in Gaussian Space". In: *Proceedings of the 31st Conference On Learning Theory*. Ed. by Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet. Vol. 75. Proceedings of Machine Learning Research. PMLR, July 2018, pp. 1887–1930. URL: https://proceedings.mlr.press/v75/dudeja18a.html.

[DLS22] Alexandru Damian, Jason Lee, and Mahdi Soltanolkotabi. "Neural Networks can Learn Representations with Gradient Descent". In: *Proceedings of Thirty Fifth Conference on Learning Theory*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 5413–5452.

[Don+23] Jonathan Dong et al. "Phase Retrieval: From Computational Imaging to Machine Learning: A tutorial". In: *IEEE Signal Processing Magazine* 40.1 (2023), pp. 45–57. DOI: 10.1109/MSP.2022.3219240[17].

[Eve+23] Mathieu Even et al. "(S)GD over Diagonal Linear Networks: Implicit Bias, Large Stepsizes and Edge of Stability". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 36. arXiv:2302.08982. 2023.

[For+21] Pierre Foret et al. "Sharpness-aware Minimization for Efficiently Improving Generalization". In: *International Conference on Learning Representations*. 2021.

[Gho+19] Behrooz Ghorbani et al. "Limitations of Lazy Training of Two-layers Neural Network". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019.

[Gho+20] Behrooz Ghorbani et al. "When Do Neural Networks Outperform Kernel Methods?" In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 14820–14830.

[Gol+19] Sebastian Goldt et al. "Dynamics of stochastic gradient descent for two-layer neural networks in the teacher-student setup". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019.

[Gol+20] Sebastian Goldt et al. "Modeling the Influence of Data Structure on Learning in Neural Networks: The Hidden Manifold Model". In: *Phys. Rev. X* 10 (4 Dec. 2020), p. 041044. DOI: 10.1103/PhysRevX.10.041044[18].

[Gol+22] Sebastian Goldt et al. "The Gaussian equivalence of generative models for learning with shallow neural networks". In: *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*. Ed. by Joan Bruna, Jan Hesthaven, and

[16] https://arxiv.org/abs/1810.04805
[17] https://doi.org/10.1109/MSP.2022.3219240
[18] https://doi.org/10.1103/PhysRevX.10.041044

Lenka Zdeborova. Vol. 145. Proceedings of Machine Learning Research. PMLR, Aug. 2022, pp. 426–471.

[Goy+17] Priya Goyal et al. "Accurate, large minibatch sgd: Training imagenet in 1 hour". In: *arXiv preprint arXiv:1706.02677* (2017).

[Gra49] Harold Grad. "Note on N-dimensional hermite polynomials". en. In: *Communications on Pure and Applied Mathematics* 2.4 (1949), pp. 325–330. ISSN: 1097-0312. DOI: 10.1002/cpa.3160020402[19].

[HC23] Karl Hajjar and Lénaïc Chizat. "On the symmetries in the dynamics of wide two-layer neural networks". In: *Electronic Research Archive* 31.4 (2023), pp. 2175–2212. ISSN: 2688-1594. DOI: 10.3934/era.2023112[20]. URL: https://www.aimspress.com/article/doi/10.3934/era.2023112.

[HW14] Moritz Hardt and Mary Wootters. "Fast matrix completion without the condition number". In: *Proceedings of The 27th Conference on Learning Theory*. Ed. by Maria Florina Balcan, Vitaly Feldman, and Csaba Szepesvári. Vol. 35. Proceedings of Machine Learning Research. Barcelona, Spain: PMLR, June 2014, pp. 638–678. URL: https://proceedings.mlr.press/v35/hardt14a.html.

[Jac+21] Arthur Jacot et al. "Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity". In: *arXiv preprint arXiv:2106.15933* (2021).

[JCH16] Kishore Jaganathan, Yonina C. Eldar, and Babak Hassibi. "Phase Retrieval: An Overview of Recent Developments". In: *Optical Compressive Imaging*. Ed. by Adrian Stern. CRC Press, 2016, pp. 263–296. ISBN: 9781315371474.

[JGH18] Arthur Jacot, Franck Gabriel, and Clement Hongler. "Neural Tangent Kernel: Convergence and Generalization in Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018.

[JMS24] Nirmit Joshi, Theodor Misiakiewicz, and Nathan Srebro. "On the Complexity of Learning Sparse Functions with Statistical and Gradient Queries". In: *Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024.

[JN15] Prateek Jain and Praneeth Netrapalli. "Fast Exact Matrix Completion with Finite Samples". In: *Proceedings of The 28th Conference on Learning Theory*. Ed. by Peter Grünwald, Elad Hazan, and Satyen Kale. Vol. 40. Proceedings of Machine Learning Research. Paris, France: PMLR, July 2015, pp. 1007–1034. URL: https://proceedings.mlr.press/v40/Jain15.html.

[JNS13] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. "Low-rank matrix completion using alternating minimization". In: *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*. 2013, pp. 665–674.

[Kor76] G. M. Korpelevič. "An extragradient method for finding saddle points and for other problems". In: *Èkonom. i Mat. Metody* 12.4 (1976), pp. 747–756. ISSN: 0424-7388.

[19] https://doi.org/10.1002/cpa.3160020402

[20] https://doi.org/10.3934/era.2023112

[Kou+24] Yiwen Kou et al. "Matching the Statistical Query Lower Bound for k-Sparse Parity Problems with Sign Stochastic Gradient Descent". In: *Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024.

[Kwo+19] Jeongyeol Kwon et al. "Global Convergence of the EM Algorithm for Mixtures of Two Component Linear Regression". In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by Alina Beygelzimer and Daniel Hsu. Vol. 99. Proceedings of Machine Learning Research. PMLR, June 2019, pp. 2055–2110. URL: https://proceedings.mlr.press/v99/kwon19a.html.

[LAL19] Wangyu Luo, Wael Alghamdi, and Yue M Lu. "Optimal spectral initialization for signal recovery with applications to phase retrieval". In: *IEEE Transactions on Signal Processing* 67.9 (2019), pp. 2347–2356.

[Lee+24] Jason D. Lee et al. "Neural network learns low-dimensional polynomials with SGD near the information-theoretic limit". In: *Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024.

[Li+20] Li Li et al. "A review of applications in federated learning". In: *Computers & Industrial Engineering* 149 (2020), p. 106854.

[Li+23] Hongkang Li et al. "A theoretical understanding of shallow vision transformers: Learning, generalization, and sample complexity". In: *arXiv preprint arXiv:2302.06015* (2023).

[Li+25] Bingrui Li et al. "On the Optimization and Generalization of Two-layer Transformers with Sign Gradient Descent". In: *The Thirteenth International Conference on Learning Representations*. 2025.

[LJ21] Chris Junchi Li and Michael Jordan. "Stochastic Approximation for Online Tensorial Independent Component Analysis". In: *Proceedings of Thirty Fourth Conference on Learning Theory*. Ed. by Mikhail Belkin and Samory Kpotufe. Vol. 134. Proceedings of Machine Learning Research. PMLR, Aug. 2021, pp. 3051–3106. URL: https://proceedings.mlr.press/v134/li21a.html.

[Lou+21] Bruno Loureiro et al. "Learning curves of generic features maps for realistic datasets with a teacher-student model". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 18137–18151.

[LT91] Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: Isoperimetry and Processes*. en. Google-Books-ID: juC1QgAACAAJ. Springer-Verlag, 1991. ISBN: 9780387520131.

[Mai+20] Antoine Maillard et al. "Phase retrieval in high dimensions: Statistical and computational phase transitions". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 11071–11082.

[Mar+24] Pierre Marion et al. "Attention layers provably solve single-location regression". In: *arXiv preprint arXiv:2410.01537* (2024).

[MB11] Eric Moulines and Francis Bach. "Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Machine Learning". In: *Advances in Neural Information Processing Systems*. Ed. by J. Shawe-Taylor et al. Vol. 24. Curran Associates, Inc., 2011.

[MB23] Pierre Marion and Raphaël Berthier. "Leveraging the two-timescale regime to demonstrate convergence of neural networks". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 64996–65029.

[MBB18] Siyuan Ma, Raef Bassily, and Mikhail Belkin. "The Power of Interpolation: Understanding the Effectiveness of SGD in Modern Over-parametrized Learning". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, July 2018, pp. 3325–3334.

[MBB24] Simon Martin, Francis Bach, and Giulio Biroli. "On the Impact of Overparameterization on the Training of a Shallow Neural Network in High Dimensions". In: *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*. Ed. by Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li. Vol. 238. Proceedings of Machine Learning Research. PMLR, May 2024, pp. 3655–3663. URL: https://proceedings.mlr.press/v238/martin24a.html.

[MM18] Marco Mondelli and Andrea Montanari. "Fundamental Limits of Weak Recovery with Applications to Phase Retrieval". In: *Proceedings of the 31st Conference On Learning Theory*. Ed. by Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet. Vol. 75. Proceedings of Machine Learning Research. PMLR, July 2018, pp. 1445–1450.

[MMM19] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. "Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit". In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by Alina Beygelzimer and Daniel Hsu. Vol. 99. Proceedings of Machine Learning Research. PMLR, June 2019, pp. 2388–2464.

[MMM22] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. "Generalization error of random feature and kernel methods: Hypercontractivity and kernel matrix concentration". In: *Applied and Computational Harmonic Analysis* 59 (2022). Special Issue on Harmonic Analysis and Machine Learning, pp. 3–84. ISSN: 1063-5203. DOI: https://doi.org/10.1016/j.acha.2021.12.003[21].

[MMN18] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. "A mean field view of the landscape of two-layer neural networks". In: *Proceedings of the National Academy of Sciences* 115.33 (2018), E7665–E7671.

[Mon+24] Behrad Moniri et al. "A theory of non-linear feature learning with one gradient step in two-layer neural networks". In: *Proceedings of the 41st International Conference on Machine Learning*. Vienna, Austria, 2024.

[Mou+23] Alireza Mousavi-Hosseini et al. "Gradient-Based Feature Learning under Structured Data". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Vol. 36. Curran Associates, Inc., 2023, pp. 71449–71485. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/e21955c93dede886af1d0d362c756757-Paper-Conference.pdf.

[21] https://doi.org/https://doi.org/10.1016/j.acha.2021.12.003

[Mou+25] Alireza Mousavi-Hosseini et al. "When Do Transformers Outperform Feedforward and Recurrent Networks? A Statistical Perspective". In: *arXiv preprint arXiv:2503.11272* (2025).

[MU25] Andrea Montanari and Pierfrancesco Urbani. "Dynamical decoupling of generalization and overfitting in large two-layer networks". In: *arXiv preprint arXiv:2502.21269* (2025).

[MUZ21] Francesca Mignacco, Pierfrancesco Urbani, and Lenka Zdeborová. "Stochasticity helps to navigate rough landscapes: comparing gradient-descent-based algorithms in the phase retrieval problem". In: *Machine Learning: Science and Technology* 2.3 (July 2021), p. 035029. DOI: 10.1088/2632-2153/ac0615[22].

[ODo14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge: Cambridge University Press, 2014. ISBN: 9781107038325. DOI: 10.1017/CBO9781139814782[23].

[Paq+22] Courtney Paquette et al. "Homogenization of SGD in high-dimensions: Exact dynamics and generalization properties". In: *arXiv preprint arXiv:2205.07069* (2022).

[Pet+22] Leonardo Petrini et al. "Learning sparse features can lead to overfitting in neural networks". In: *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*. Ed. by Sanmi Koyejo et al. 2022. URL: http://papers.nips.cc/paper%5C_files/paper/2022/hash/3d3a9e085540c65dd3e5731361f9320e-Abstract-Conference.html.

[Pfl86] Georg Ch. Pflug. "Stochastic Minimization with Constant Step-Size: Asymptotic Laws". In: *SIAM Journal on Control and Optimization* 24.4 (1986), pp. 655–666. DOI: 10.1137/0324039[24]. eprint: https://doi.org/10.1137/0324039.

[PPF21] Scott Pesme, Loucas Pillaud-Vivien, and Nicolas Flammarion. "Implicit Bias of SGD for Diagonal Linear Networks: a Provable Benefit of Stochasticity". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 29218–29230.

[RB95] P Riegler and M Biehl. "On-line backpropagation in two-layered neural networks". In: *Journal of Physics A: Mathematical and General* 28.20 (Oct. 1995), pp. L507–L513. DOI: 10.1088/0305-4470/28/20/002[25].

[RBG25] Fabiola Ricci, Lorenzo Bardone, and Sebastian Goldt. "Feature learning from non-Gaussian inputs: the case of Independent Component Analysis in high dimensions". In: *Proceedings of the 42nd International Conference on Machine Learning*. Ed. by Aarti Singh et al. Vol. 267. Proceedings of Machine Learning Research. PMLR, July 2025, pp. 51614–51639. URL: https://proceedings.mlr.press/v267/ricci25a.html.

[Ref+21] Maria Refinetti et al. "Classifying high-dimensional Gaussian mixtures: Where kernel methods fail and neural networks succeed". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang.

[22] https://doi.org/10.1088/2632-2153/ac0615
[23] https://doi.org/10.1017/CBO9781139814782
[24] https://doi.org/10.1137/0324039
[25] https://doi.org/10.1088/0305-4470/28/20/002

Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 8936–8947.

[Ric+23] Frederieke Richert et al. "Layered Neural Networks with GELU Activation, a Statistical Mechanics Analysis." In: *ESANN*. 2023.

[RM51] Herbert Robbins and Sutton Monro. "A Stochastic Approximation Method". In: *The Annals of Mathematical Statistics* 22.3 (1951), pp. 400–407. DOI: 10.1214/aoms/1177729586[26].

[Rob16] Elina Robeva. "Orthogonal Decomposition of Symmetric Tensors". In: *SIAM Journal on Matrix Analysis and Applications* 37.1 (2016), pp. 86–102.

[RU98] G. Reents and R. Urbanczik. "Self-Averaging and On-Line Learning". In: *Phys. Rev. Lett.* 80 (24 June 1998), pp. 5445–5448. DOI: 10.1103/PhysRevLett.80.5445[27].

[RV22] Grant Rotskoff and Eric Vanden-Eijnden. "Trainability and Accuracy of Artificial Neural Networks: An Interacting Particle System Approach". In: *Communications on Pure and Applied Mathematics* 75.9 (2022), pp. 1889–1935. DOI: https://doi.org/10.1002/cpa.22074[28]. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.22074. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.22074.

[Sar+20] Stefano Sarao Mannelli et al. "Complex Dynamics in Simple Neural Networks: Understanding Gradient Flow in Phase Retrieval". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 3265–3274.

[SHT23] Clayton Sanford, Daniel J Hsu, and Matus Telgarsky. "Representational strengths and limitations of transformers". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 36677–36707.

[SMG14] A Saxe, J McClelland, and S Ganguli. "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks". In: *Proceedings of the International Conference on Learning Represenatations 2014*. International Conference on Learning Represenatations 2014. 2014.

[Son+24] Bingqing Song et al. "Unraveling the gradient descent dynamics of transformers". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 92317–92351.

[SQW16] Ju Sun, Qing Qu, and John Wright. "A geometric analysis of phase retrieval". In: *2016 IEEE International Symposium on Information Theory (ISIT)*. 2016, pp. 2379–2383. DOI: 10.1109/ISIT.2016.7541725[29].

[SS20] Justin Sirignano and Konstantinos Spiliopoulos. "Mean field analysis of neural networks: A central limit theorem". In: *Stochastic Processes and their Applications* 130.3 (2020), pp. 1820–1852.

[26] https://doi.org/10.1214/aoms/1177729586
[27] https://doi.org/10.1103/PhysRevLett.80.5445
[28] https://doi.org/https://doi.org/10.1002/cpa.22074
[29] https://doi.org/10.1109/ISIT.2016.7541725

[SS95a] David Saad and Sara A. Solla. "Exact Solution for On-Line Learning in Multilayer Neural Networks". In: *Phys. Rev. Lett.* 74 (21 May 1995), pp. 4337–4340. DOI: 10.1103/PhysRevLett.74.4337[30].

[SS95b] David Saad and Sara A. Solla. "On-line learning in soft committee machines". In: *Phys. Rev. E* 52 (4 Oct. 1995), pp. 4225–4243. DOI: 10.1103/PhysRevE.52.4225[31].

[SS96] David Saad and Sara Solla. "Dynamics of On-Line Gradient Descent Learning for Multilayer Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by D. Touretzky, M. C. Mozer, and M. Hasselmo. Vol. 8. MIT Press, 1996.

[SVZ20] Stefano Sarao Mannelli, Eric Vanden-Eijnden, and Lenka Zdeborová. "Optimization and Generalization of Shallow Neural Networks with Quadratic Activation Functions". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 13445–13455.

[SW24] Antonio Sclocchi and Matthieu Wyart. "On the different regimes of stochastic gradient descent". In: *Proceedings of the National Academy of Sciences* 121.9 (2024), e2316301121. DOI: 10.1073/pnas.2316301121[32]. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2316301121. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2316301121.

[TKB20] Matthew Trager, Kathlén Kohn, and Joan Bruna. "Pure and Spurious Critical Points: A Geometric Study of Linear Networks". In: *International Conference on Learning Representations (ICLR)*. arXiv:1910.01671. 2020.

[Tro+25a] Emanuele Troiani et al. "Fundamental computational limits of weak learnability in high-dimensional multi-index models". In: *The 28th International Conference on Artificial Intelligence and Statistics*. 2025.

[Tro+25b] Emanuele Troiani et al. "Fundamental limits of learning in sequence multi-index models and deep attention networks: High-dimensional asymptotics and sharp thresholds". In: *arXiv preprint arXiv:2502.00901* (2025).

[TV18] Yan Shuo Tan and Roman Vershynin. "Phase retrieval via randomized Kaczmarz: theoretical guarantees". In: *Information and Inference: A Journal of the IMA* 8.1 (Apr. 2018), pp. 97–123. ISSN: 2049-8772. DOI: 10.1093/imaiai/iay005[33].

[TV23] Yan Shuo Tan and Roman Vershynin. "Online Stochastic Gradient Descent with Arbitrary Initialization Solves Non-smooth, Non-convex Phase Retrieval". In: *Journal of Machine Learning Research* 24.58 (2023), pp. 1–47. URL: http://jmlr.org/papers/v24/20-902.html.

[Vas+17] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[VBS19] Sharan Vaswani, Francis Bach, and Mark Schmidt. "Fast and Faster Convergence of SGD for Over-Parameterized Models and an Accelerated Perceptron". In: *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and*

[30] https://doi.org/10.1103/PhysRevLett.74.4337
[31] https://doi.org/10.1103/PhysRevE.52.4225
[32] https://doi.org/10.1073/pnas.2316301121
[33] https://doi.org/10.1093/imaiai/iay005

Statistics. Ed. by Kamalika Chaudhuri and Masashi Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, Apr. 2019, pp. 1195–1204.

[Vei+22] Rodrigo Veiga et al. "Phase diagram of Stochastic Gradient Descent in high-dimensional two-layer neural networks". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 23244–23255. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/939bb847ebfd14c6e4d3b5705e562054-Paper-Conference.pdf.

[Ver18] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge: Cambridge University Press, 2018. ISBN: 9781108415194. DOI: 10.1017/9781108231596[34]. (Visited on 11/24/2022).

[VPF21] Aditya Vardhan Varre, Loucas Pillaud-Vivien, and Nicolas Flammarion. "Last iterate convergence of SGD for Least-Squares in the Interpolation regime." In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 21581–21591.

[VW96] Aad van der Vaart and Jon Wellner. *Weak Convergence and Empirical Processes: With Applications to Statistics*. en. Google-Books-ID: OCenCW9qmp4C. Springer Science & Business Media, Mar. 1996. ISBN: 9780387946405.

[Wan+24] Zixuan Wang et al. "Transformers provably learn sparse token selection while fully-connected nets cannot". In: *arXiv preprint arXiv:2406.06893* (2024).

[WJ22] Xiaoyu Wang and Mikael Johansson. *On Uniform Boundedness Properties of SGD and its Momentum Variants*. Tech. rep. arXiv:2201.10245 [cs, math] type: article. arXiv, June 2022. DOI: 10.48550/arXiv.2201.10245[35].

[Wu+23] Yongtao Wu et al. "On the convergence of encoder-only shallow transformers". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 52197–52237.

[Yan19] Greg Yang. "Tensor Programs I: Wide Feedforward or Recurrent Neural Networks of Any Architecture are Gaussian Processes". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 32. arXiv:1910.12478. 2019.

[YF25] Oğuz Kaan Yüksel and Nicolas Flammarion. "On the Sample Complexity of Next-Token Prediction". In: *The 28th International Conference on Artificial Intelligence and Statistics*. 2025.

[YO19] Yuki Yoshida and Masato Okada. "Data-dependence of plateau phenomenon in learning with neural network—Statistical mechanical analysis". In: *Advances in Neural Information Processing Systems* 32 (2019).

[Yos+17] Yuki Yoshida et al. "Statistical Mechanical Analysis of Online Learning with Weight Normalization in Single Layer Perceptron". In: *Journal of the Physical Society of Japan* 86.4 (Apr. 2017), p. 044002. ISSN: 0031-9015. DOI: 10.7566/JPSJ.86.044002[36].

[ZB24] Aaron Zweig and Joan Bruna. "Symmetric Single Index Learning". In: *The Twelfth International Conference on Learning Representations*. 2024.

[34] https://doi.org/10.1017/9781108231596

[35] https://doi.org/10.48550/arXiv.2201.10245

[36] https://doi.org/10.7566/JPSJ.86.044002

[Zen20] Caibin Zeng. "Mean exit time and escape probability for the Ornstein–Uhlenbeck process". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 30.9 (2020), p. 093127.

[Zha+19] Michael Zhang et al. "Lookahead Optimizer: k steps forward, 1 step back". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/90fd4f88f588ae64038134f1eeaa023f-Paper.pdf.

[Zha+25] Yingying Zhang et al. "Understanding Generalization in Transformers: Error Bounds and Training Dynamics Under Benign and Harmful Overfitting". In: *arXiv preprint arXiv:2502.12508* (2025).

[Zou+21] Difan Zou et al. "Benign Overfitting of Constant-Stepsize SGD for Linear Regression". In: *Proceedings of Thirty Fourth Conference on Learning Theory*. Ed. by Mikhail Belkin and Samory Kpotufe. Vol. 134. Proceedings of Machine Learning Research. PMLR, Aug. 2021, pp. 4633–4635.

[ZPB23] Aaron Zweig, Loucas Pillaud-Vivien, and Joan Bruna. "On single-index models beyond gaussian data". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 10210–10222.

Curriculum Vitae

Personal Information

Name	Luca Arnaboldi
Location	Lausanne, Switzerland
Email	luca@arnaboldi.lu
Website	www.arnaboldi.lu
Google Scholar	`https://scholar.google.com/citations?user=A-4QdoQAAAAJ`
GitHub	github.com/arn4

Experience

Applied Scientist Intern @ Amazon Web Services (AWS) — Aug 2025 – Dec 2025
AWS AI Fundamental Research team. — Pasadena, United States

Education

PhD in Theory of Machine Learning @ EPFL — Oct 2022 – Present
Supervisor: Prof. Florent Krzakala. — Lausanne, Switzerland

Diploma in Physics @ Scuola Normale Superiore — Oct 2017 – Oct 2022
~30 students admitted per year. Grade: 100/100 cum laude. — Pisa, Italy

MSc in Complex Systems @ University of Pisa — Sep 2020 – Sep 2022
Grade: 110/110 cum laude. GPA: 29.7/30.0. — Pisa, Italy

BSc in Physics @ University of Pisa — Sep 2017 – Sep 2020
Grade: 110/110 cum laude. GPA: 29.6/30.0. — Pisa, Italy

Awards

Top Reviewer @ NeurIPS 2025 2025

Finalist @ Codeflows 2021 2021
Coding competition by Bending Spoons – 26th of ~3400.

Bronze Medal @ ICPC SWERC 2021 2021
Scuola Normale Superiore team.

Silver Medal @ International Physics Olympiad (IPhO) 2017
Yogyakarta, Indonesia

1st place @ Italian Team Olympiad in Informatics 2016 2016
Bologna, Italy

Conferences, Schools & Workshops

Simons Collaboration Meeting on Physics of Learning July 2026
Short Talk Taipei, Taiwan

International Conference on Machine Learning (ICML) July 2026
Satellite Talk Seoul, South Korea

LemanTh May 2026
Poster Lausanne, Switzerland

Workshop on Late Interaction Models April 2026
Poster Delft, Netherlands

European Conference on Information Retrieval (ECIR) April 2026
- Delft, Netherlands

Conference on Neural Information Processing Systems (NeurIPS) December 2025
Poster San Diego, United States

STATPHYS 29 July 2025
Poster Florence, Italy

LemanTh May 2025
Poster Lausanne, Switzerland

ROCKING AI v1 September 2024
Poster Roccella Ionica, Italy

Workshop on High-dimensional Learning Dynamics July 2024
Poster Vienna, Austria

International Conference on Machine Learning (ICML) July 2024
2 Posters Vienna, Austria

Deep Learning and Reinforcement Learning Summer School June 2024
Poster, Vector Institute Toronto, Canada

Mathematics of Machine Learning May 2024
Poster Zurich, Switzerland

LemanTh May 2024
Poster Lausanne, Switzerland

Workshop on the Challenges in the Theory of Neural Networks Dynamics October 2023
- Paris, France

Visit at ENS Ulm hosted by B. Loureiro September 2023
- Paris, France

Statistical Physics and Machine Learning Back Together Again August 2023
Poster Cargese, France

Conference on Learning Theory (COLT) July 2023
Short Talk Bangalore, India

Princeton Machine Learning Theory Summer School June 2023
Poster Princeton, United States

Machine Learning Crash Course 2022 @ UniGe June 2022
- Genoa, Italy

Open Source Contributions

MLX: Apple Silicon library for ML. 6 PRs merged

Sentence-Transformers: Python library for sentence embeddings. 1 PR merged

PyTorch: Deep learning framework. 1 PR merged

PyLate: Python library for late interaction models. 2 PRs merged

Scientific Reviewing

NeurIPS: Conference 2025

ICLR: Conference 2025

Physical Review E: Journal 2025

NeurIPS: Conference 2024

AISTATS: Conference 2024

Teaching & Supervision Experience

Teaching Assistant: EE-452: Network Machine Learning Spring 2026

Teaching Assistant: EE-411: Fundamentals of Inference and Learning Autumn 2023 & 2024

Semester Project: RAG at/for Idephics – K.L., R.G. Autumn 2024

Teaching Assistant: EE-209: Éléments de Statistiques pour les Data Sciences Spring 2024

Semester Project: Group Invariance for Convolutional Neural Networks – M.Z. Autumn 2023

Semester Project: Group Convolutional Neural Networks – D.A. Autumn 2023

Teaching Assistant: MGT-466: Negotiation Techniques Spring 2023

Publications

1. **Humanity's Last Exam: A benchmark of expert-level academic questions to assess AI capabilities** Center for AI Safety, Scale AI, HLE Contributors Consortium. Nature, vol. 649, 2026.
2. **ColBERT-Zero: To Pre-train Or Not To Pre-train ColBERT models** A Chaffin, L Arnaboldi, A Chatelain, F Krzakala. Late Interaction Workshop at ECIR, 2026.
3. **Asymptotics of SGD in Sequence-Single Index Models and Single-Layer Attention Networks** L Arnaboldi, B Loureiro, L Stephan, F Krzakala, L Zdeborová. Neural Information Processing Systems (NeurIPS), 2025.
4. **The Benefits of Reusing Batches for Gradient Descent in Two-Layer Networks: Breaking the Curse of Information and Leap Exponents** Y Dandi, E Troiani, L Arnaboldi, L Pesce, L Zdeborová, F Krzakala. International Conference on Machine Learning (ICML), 2024.
5. **Online Learning and Information Exponents: On The Importance of Batch size, and Time / Complexity Tradeoffs** L Arnaboldi, Y Dandi, F Krzakala, B Loureiro, L Pesce, L Stephan. International Conference on Machine Learning (ICML), 2024.
6. **From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layer networks** L Arnaboldi, L Stephan, F Krzakala, B Loureiro. Conference on Learning Theory (COLT), 2023.

Preprints

1. **Deep Learning as Neural Low-Degree Filtering: A Spectral Theory of Hierarchical Feature Learning** Y Dandi, M Vilucchio, L Arnaboldi, H Tabanelli, F Krzakala. arXiv preprint, under review at NeurIPS 2026, 2026.
2. **Repetita Iuvant: Data Repetition Allows SGD to Learn High-Dimensional Multi-Index Functions** L Arnaboldi, Y Dandi, F Krzakala, L Pesce, L Stephan. arXiv preprint, 2024.
3. **Escaping mediocrity: how two-layer networks learn hard generalized linear models with SGD** L Arnaboldi, F Krzakala, B Loureiro, L Stephan. arXiv preprint, accepted at NeurIPS 2023 Workshop OPT, 2023.