



UNIVERSITATEA DIN
BUCUREȘTI
VIRTUTE ET SAPIENTIA



FACULTATEA DE FIZICĂ

PHASE TRANSITIONS IN A TOY MODEL OF SUPERPOSITION

MASTER'S DISSERTATION

Author

Homeucă Roberto-Paul

Scientific supervisors

Dr. Călin-Iuliu Lăzăroiu
Prof. univ. dr. Virgil Băran

Bucharest, 2026

Acknowledgements

First and foremost, I would like to express my gratitude to my supervisor, Călin Lăzăroiu, who introduced me to this subject and guided me throughout. Beyond the science, his breadth of knowledge — from psychology to much else besides — genuinely broadened my horizons, and his fairness and readiness to help have made him a model of what a scientist should be.

I am also deeply grateful to my family and my partner, whose support and encouragement were a constant source of motivation: my mother, Paulina Belciu, my brother, Albert Homeucă, and my partner, Petronela Crețu — and not least our dog, Sven, who always knew when I needed a walk.

Contents

1	Introduction and overview	1
2	Machine learning, physics and AI safety	5
2.1	A brief history of AI	5
2.2	The “black box” approach: behaviorism and machine learning	7
2.3	The influence of physics on AI	8
2.4	A historical primer on AI safety	9
2.5	Alignment & interpretability: why the “black box” must be opened	11
2.6	Opening the “black box” using tools from physics	12
3	Machine learning preliminaries	15
3.1	Introductory terminology	15
3.2	The McCulloch-Pitts neuron	17
3.3	The mathematical model of ANNs	18
3.4	Context on statistical learning theory	21
4	Singular Learning Theory	23
4.1	Bayesian statistics	24
4.1.1	The framework of parameterized Bayesian inference	24
4.1.2	Core quantities of SLT and objectives	25
4.1.3	The partition function and free energy	28
4.2	Regular vs. singular models	29
4.3	The machinery of SLT	31
4.4	Bayesian phase transitions	36
5	Toy models of superposition	39
5.1	A primer on superposition in ANNs	39
5.2	Introduction of TMS	40
5.3	SLT analysis for the case $r = 2$	42
5.4	Study of the case $r = 1$	44
5.4.1	The case $c = k$ with $b_i \leq 0$	46
5.4.2	The case of negative bias for $k < c$	51
5.4.3	The case of positive bias on vestigial columns	55
5.4.4	Hessian directional analysis	57
	The case of negative bias	58
	The case of positive bias	60
5.5	Experiments which confirm our results for the case $r = 1$	61
5.5.1	Random init	61
5.5.2	$(1, 1)$ + standard vestigials	62
5.5.3	$(1, 1) + \sigma$ - positive vestigials (full)	62
5.5.4	$(1, 0)$ trap	63
5.5.5	All-same-sign saddle	64
5.5.6	Remarks	64
6	Conclusions	67
	References	69

Introduction and overview

The field of artificial intelligence (AI) gained remarkable momentum and public attention in recent years, after having languished largely outside of public view during the slow “thaw” of 1993-2012 that followed the second AI winter (1987-1993). Unlike previous AI springs, the current wave of optimism relies on progress in artificial neural networks (ANNs), namely advances in deep neural networks (DNNs, defined as those ANNs which contain more than one ¹ “hidden layer” between the input and output layer) and their applications to machine learning and generative AI.

Contemporary developments trace back to the success achieved by AlexNet (a deep convolutional neural network developed by Alex Krizhevsky, Ilya Sutskever and Geoffrey Hinton, see [1]) in the 2012 ImageNet Large Scale Visual Recognition Challenge (<https://www.image-net.org/challenges/LSVRC>). This and subsequent developments (such as the encoder-decoder and transformer architectures developed in [2, 3], the release of the GPT-3 foundation model [4] and later of ChatGPT, the breakthroughs achieved by Deep Mind with AlphaGo, AlphaZero and AlphaFold [5, 6, 7, 8, 9, 10, 11, 12] and the recent proliferation of reasoning, multimodal and agentic models) were in large part driven by the explosive increase in the computational power of modern GPUs, the use of massive labeled datasets extracted from internet content and a number of new algorithmic techniques which led to fast progress in DNNs. We mention that the history of ANNs traces back to the 1940s (see the initial papers [13, 14] and the historical accounts in [15, 16, 17, 18, 19]), while DNNs trace back to the 1980s (see the historical accounts in [20, 21]). We refer the reader to [22] for a comprehensive introduction to the field of artificial intelligence and to [23, 24, 25, 26, 27] for comprehensive treatments of deep learning. A conceptual but non-technical overview of the subject can be found in [28]. See [29, 30, 31, 32] for background on the wider subject of machine learning.

Since sufficient computational power has become available and relatively affordable to implement and train very large scale DNNs (a fact which stimulated massive investment in hyperscalers), new methods and technologies have emerged which implement the underlying theory and previous work in this field at scale. This led to the current AI boom [33, 34, 35], a period of explosive growth popularized initially by large language models (LLMs), followed by other forms of generative [36, 37] and agentic [38, 39] AI. See [40, 41, 42, 43] for an introduction to LLMs, which are the tools most widely used at present by the general population.

This fast-developing field is transforming our digital ecosystem and has increasingly

¹Current frontier LLMs often have between 48-120 transformer-based hidden layers, with layer width/hidden dimension around or over 10,000 and use various optimized architectures such as Mixture of Experts, often having over 1 trillion parameters.

profound economic, organizational, social, cultural, political and psychosocial impacts [44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56] through its institutional, industrial and commercial implementation and its effects on networked communication, knowledge acquisition and feedback systems used in everyday life (AI-enabled online commerce and social media, agentic browsers, AI assistants and companions, algorithmic recommender systems and so on).

Given how aggressively such systems are being rolled out and implemented, one might naively expect that the underlying behavior of this technology is well-understood, explainable, predictable, reliable and under tight control. However, this assumption is incorrect. Given the high complexity and scale of these systems, massive DNNs largely function as “black boxes” in how and what they learn and what internal representations they develop to encode and process information — a fact which currently precludes us from reliably interpreting or predicting their behavior at scale. In fact, current training methods based on supervised and unsupervised reinforcement learning trace back to mathematical models initially developed by behavioral psychologists during the 1970s, since the theory of reinforcement learning originated from the Skinnerian paradigm of operant conditioning [57, 58, 59]. As such, training a neural network is akin to training an animal or child through behavioral conditioning — a method which is well-known to be prone to occasional but sometimes major and unexpected failure. The problem is compounded by the fact that DNNs *do not* function like animals or children, with whom we are lucky to have previous and extensive experience at societal scale. Rather, they often behave in surprising and unexpected ways which keep changing and becoming ever more complex as technology keeps developing exponentially.

Given the various risks inherent in increasingly relying on systems whose behavior we don’t fully comprehend and control, increasingly great focus has been placed during the past few years on understanding the inner works and effective control parameters of such systems, as fast progress on the development and implementation side outgrew our understanding of their behavior. The larger and more complex such systems become, the harder they are to interpret, predict and align with human intent, which amplifies the potential for intentional misuse or accidental misalignment (see [60, 61, 62, 63, 64, 65, 66, 67, 68, 69]). Consequently, we find ourselves in a situation in which highly complex and impactful AI systems are being built and deployed at breakneck speed in an extremely competitive global economical and geopolitical landscape, yet their internal workings remain poorly understood and their behavior is hard to fully analyze, interpret, predict and control.

The field of **AI safety** has emerged to address this problem, aiming to reduce and limit the risk that an AI system might behave in an unpredictable, unintended, malicious or deceptive manner. This led to a number of international summits (Bletchley Park 2023, Seoul 2024, Paris 2025) focused on AI governance [70, 71, 72, 73, 74, 75] and to the creation of an international network of national AI safety institutes and other organizations which report annually on the issue (see [76, 77]).

We stress that fundamental problems in the subject of AI safety demand different concepts and approaches from the currently dominant applied engineering framework, precisely because they require abandoning the “black box” paradigm of behavioral conditioning which governs the latter. This is somewhat akin in difficulty to developing a *workable*, *testable* and *verifiable* “theory of mind” and a similarly strong *cognitive* theory of learning

as opposed to relying on purely behavioral models, which treat natural or artificial minds as black boxes without asking about their inner workings. The hope of achieving this for artificial minds is considerably greater than for natural minds given that — unlike human or complex animal brains and neurological systems — we know and can access the hardware specification and details of artificial neural networks and have wide freedom in experimenting on them though the implementation of various learning algorithms and varying the input data.

A useful scientific approach to this problem demands high rigor and a precise conceptual framework which allows one to extract insights through mathematical models which we construct and analyze — and ultimately for such insights to be validated by systematic experiments conducted on large scale AI systems based on DNNs.

One rigorous framework which has been growing recently in popularity for formulating and addressing the fundamental problems of AI safety is inspired directly by Statistical Physics and modern Statistical Learning Theory — in the fully general mathematical formulation of the latter, which is known as Singular Learning Theory (SLT) [78, 79]. Modern AI systems use large scale deep neural networks whose learning process can be modeled statistically, the reinforcement learning and generation process being described by SLT. In this context, Statistical Physics turns out (see [80]) to be a natural framework from which we can extract ideas, models and methods that can be generalized, modified and adapted in order to gain a better understanding of the the behavior of DNNs and to attack the problem of finding effective solutions to fundamental AI safety problems such as explainability, interpretability and scalable alignment. This motivates our interest in this subject which, at its deeper and most advanced levels (which we will not attempt to address in this thesis) is related to profound problems in how the theory of RG flows can be applied to complex and heterogeneous statistical systems with nonlinear long range interactions.

This thesis is structured as follows:

- **Chapter 2** contains a brief history of how the broad field of artificial intelligence evolved by drawing from many areas of science, thus demonstrating its highly interdisciplinary nature. We then describe the currently fragmented and fast-evolving field of AI safety and explain why the open problems in that subject naturally lead to a physics-based methodology grounded in SLT. Finally, we briefly review some of the literature that bridges these two disciplines.
- **Chapter 3** explains some terminology specific to the field of artificial intelligence, while defining and exemplifying the theoretical concepts and systems studied later in the thesis.
- **Chapter 4** is a review of Singular Learning Theory (SLT) [78, 79], which was gradually developed by the mathematician Sumio Watanabe starting in the mid 1990s. SLT provides the correct mathematical framework of statistical learning theory for the generic case of unidentifiable Bayesian learning models whose Fisher information metric fails to be strictly positive definite. It is grounded in Bayesian statistics [81, 82, 83, 84] and singularity theory [85, 86, 87, 88, 89] and relates in profound ways to ideas of Statistical Physics. SLT affords a mathematically rigorous

understanding of the statistical models that underpin machine learning in modern artificial intelligence systems. This theory also forms the correct mathematical framework for studying a wide range of data analysis and statistical inference problems which occur in numerous fields of experimental and observational science (such as high energy experimental physics, observational astrophysics, atmospheric and earth sciences/earth system science, biology, sociology, psychometrics etc.), where the Bayesian framework [81, 90] has become dominant (see, for example, [91, 92, 93, 94, 95, 96, 97, 98]).

- **Chapter 5** is the core of the thesis, being concerned with the special class of models which we study in detail — the so called Toy Models of Superposition (TMS). which were introduced in [99]. Such models exhibits a rich structure and complex behavior, which illustrates the phase transitions of neural networks predicted on abstract grounds by SLT. We study these model using SLT and discuss the Bayesian Antecedent Hypothesis in this context. Unlike the work of Chen et al. (see [100]), who studied the problem in two dimensions, we study such systems in one dimension, proving rigorously that in this case we cannot have true superposition.

Summary. The main objective of this thesis is to apply Singular Learning Theory, a rigorous framework for Bayesian learning related to statistical physics, to a simple toy model and study its behavior in detail. Ultimately, this work contributes to the broader ongoing effort aimed at constructing a mathematically rigorous, physics-inspired framework which would allow one to solve hard problems of interpretability, predictability and scalable alignment in the field of AI safety. The results of the thesis complete part of the picture for TMS networks, thus serving as a stepping stone toward the analysis of more complex cases which promise a deeper understanding of neural network behavior in AI safety research.

Machine learning, physics and AI safety

2.1 A brief history of AI

The primary goal of Artificial Intelligence is to build artificial systems capable of performing tasks and solving problems typically associated with human intelligence. Rather than manually programming explicit rules to solve a problem, modern artificial intelligence involves developing algorithms that learn, reason, and make decisions based on self-improvement to find optimal or nearly optimal solutions.

The field was established formally after the famous Dartmouth workshop held in 1956 and organized by John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon. The name “artificial intelligence” was coined in McCarthy’s funding proposal for the workshop (see [101]). Although this is considered the “official” birth of AI as a distinct subject, substantial work on related problems had been done before. Such work can be exemplified by Alan Turing’s early ideas on machine intelligence, which he published in his famous 1950 paper “Computing Machinery and Intelligence” [102] or the 1943 research by the neurophysiologist Warren McCulloch and the logician Walter Pitts (see [13]), who proposed the first mathematical model of an artificial neuron (the *MCP neuron*), inspired by how real neurons operate. They also showed that an (infinite) directed graph of artificial neurons could compute any computable function by showing that such a graph is computationally equivalent with a universal Turing machine. Besides contributing to the birth of artificial neural networks, their paper initiated the field of computational neuroscience.

Before saying more about artificial neurons and ANNs, let us briefly describe the evolution of the field after the Dartmouth workshop, following the two main directions already considered there: *symbolic AI* (based on logic and deductive reasoning) and *connectionism* (based on neural networks).

The Dartmouth workshop initiated the **first AI spring** — a period of fast progress, increasing interest and serious funding from US agencies such as ARPA (the predecessor to DARPA). This period focused mainly on the symbolic approach and ended around 1974, in large part due to the lack of military breakthroughs (which disappointed DARPA¹ and the DoD) and of the Lighthill report [103], which heavily criticized the field relying, among others, on the argument of the “combinatorial explosion” while claiming that the pursuit of AGI and of advanced robotics was a dead end and that AI had no identity as a stand alone scientific subject. During the first AI spring, important progress was made on:

¹ARPA was renamed to DARPA in 1972.

- Search and heuristics (search trees, the General Problem Solver [104] and the A^* algorithm [105]).
- Rules-based natural language processing using dictionaries and syntax (influenced by Chomsky’s theory of generative grammars [106]), which produced early chatbots such as ELIZA (1966, see [107]) and PARRY (1971, see [108]).
- AI-oriented programming, through the introduction of LISP [109].
- On the connectionist side, the introduction and study of generalized artificial neuron models such as the Perceptron [110]. The latter was analyzed mathematically by Minsky and Papert, who proved the limitations of single layer ANNs [111].

Despite the sudden cut in funding and the shift from excessive optimism to extreme pessimism, progress in AI continued during the period 1974-1980 (the **first AI winter**), leading to the birth of expert systems and knowledge engineering [112, 113, 114] as well as of logic programming² [115, 116], which was implemented by programming languages such as PROLOG [117, 118].

The first AI winter was followed by the so-called “Expert systems era” (1980-1987), a period of intense commercial activity and government funding (in particular in Japan), sometimes called **the second AI spring**. This led to mainstreaming of knowledge engineering, which continues to have a major impact on how business, governance and various organizations are being run nowadays. The commercial hype typical of this period ended due to scaling issues, hardware limitations, the famous “knowledge acquisition bottleneck” and the gradual rise of machine learning, which provided a superior alternative to expert systems.

The second AI spring was followed by the **second AI winter** (1987-1993) and by the two decade-long slow “thaw” of 1993-2012, leading to the **third AI spring** (2013-present). We end this brief historical account by listing the main developments that took place during the period 1987-2012, which gradually moved the field into its present stage:

- 1988. Increasing reliance on statistical and probabilistic models, such as the Bayesian networks proposed in [119].
- 1995. The introduction of support vector machines (SVMs) [120].
- 1997. The Long Short-Term Memory ANN architecture proposed in [121], which solved the vanishing gradient problem of classical recurrent neural networks (RNNs), leading to important progress in speech recognition, translation and time-series forecasting. The same year, IBM Deep Blue defeated Gary Kasparov (see [122]).
- 2009. The launch of ImageNet (see [123]), a massive database of labeled high-resolution images used for training and benchmarking convolutional neural networks used in the field of AI vision.

²We mention that logic programming is of increasing importance nowadays, not only for formal verification, computer security and scheduling/logistics but also through increasingly important applications to the field of AI safety — for example, in safety/legal compliance verification for critical applications of ANNs (medical, automated driving etc.) in the emerging field of neurosymbolic AI.

- The period culminated in the breakthrough achieved by AlexNet in the 2012 ImageNet competition, which initiated the current AI spring already described in the introduction.

2.2 The “black box” approach: behaviorism and machine learning

As already mentioned in the introduction, modern machine learning (ML) algorithms rely on mathematical models of reinforcement learning [124], which originated in behavioral psychology (see [125, 126, 127, 128, 129, 130, 131] for modern treatments of the latter). The connection between the behavioral psychology framework of operant conditioning, the mathematical models of learning it inspired and modern reinforcement learning (RL) algorithms used in ML is one of the most important intellectual lineages in the subject of artificial intelligence. Awareness of this connection is important for a proper conceptual understanding of the nature and shortcomings of modern methods used to train deep neural networks. Much of modern reinforcement learning as applied in ML is a formalization and generalization of ideas that originated in behavioral psychology. This helps clarify conceptually the fundamental limitations of current deep learning paradigms from the viewpoint of AI safety.

The theory of reinforcement learning in behavioral psychology formalizes trial-and-error learning via rewards and punishments, using update rules similar to those employed in machine learning. Psychological reinforcement theories inspired early AI researchers to think of machine intelligence as emerging from interactions of a machine with its environment which are guided by rewards — rather than from explicit programming. This led to mathematical models of RL used for machine learning, based on Markov decision processes and algorithms that maximize cumulative reward. Conversely, computational models are used to analyze, simulate and model phenomena in neuroscience and behavioral psychology. Such models employ ML methods such as Q-learning, actor-critic, or SARSA-like algorithms (see [124]). Here is a brief timeline of the relevant history:

- **1930s–1950s.** The behaviorist current in psychology led to the discovery of empirical laws of learning (Skinner, Hull, Bush & Mosteller).
- **1950s–1980s.** Mathematical psychologists proposed quantitative models of those laws (Rescorla & Wagner, Herrnstein).
- **1980s–1990s.** The development of temporal-difference and optimal control theory, which connected machine learning and dynamic programming (Sutton, Watkins).
- **1990s–present.** Modern reinforcement learning theory unified previous developments into a common framework [132, 133, 134, 135]. Moreover, computational neuroscientists showed that reward prediction errors resemble dopaminergic signaling (Schultz).

In [58, 57] (see also [59]), B. F. Skinner introduced the conceptual paradigm which led to the RL framework. Hull’s book [136] proposed the first mathematical model, which

described RL through iterative updates of habit strength. In [137], Bush and Mosteller proposed the stochastic linear reward-operator model, which describes the probability of a response through a linear iteration dependent on reinforcement outcome. The paper [138] introduced the Rescorla-Wagner model, which is governed by the temporal difference error (modern temporal difference learning in ML is an extension of this model). Herrnstein’s paper [139] described the matching law, which is foundational in reward allocation and decision theory.

In 1988, Richard Sutton proposed the temporal difference learning model (see [140]), which extends the Rescorla–Wagner model by allowing learning from future predictions. In 1989, Christopher Watkins introduced Q-learning [141]. In 1997, the neuroscience paper [142] found that dopaminergic neurons encode something similar to temporal difference error.

The classic 1998 book of Sutton and Barto [143] unified operant conditioning, dynamic programming, stochastic control and machine learning into an overarching framework which forms the basis of modern RL in neural networks. Finally, the modern deep RL era starts with the 2015 paper [144] (authored by a Deep Mind collaboration), which introduced Deep Q-Networks.

2.3 The influence of physics on AI

The methods used in artificial intelligence can be traced back to mathematical principles such as linear regression and the chain rule, which are used as operational building blocks at the lowest levels of functionality (see [145]). As mentioned in the introduction and illustrated by our brief account of its history, the field of AI was rooted since its inception in mathematics as well as in theoretical computer science, biology and psychology.

Building further on the influence of different sciences on the development of the subject, let us consider briefly how physics changed the course of progress. Instances of this can be seen through how different architectures were built based on various physical systems and on ideas borrowed from physics. Such influences can be seen in Hopfield networks [146] which were modeled after systems of magnetic spins in condensed matter systems and introduced an “energy function”, which, when minimized, would drive the network to its objective. Based on these, David H. Ackley, Geoffrey Hinton and Terry Sejnowski introduced Boltzmann machines [147] which are rooted in statistical mechanics — thus introducing the concept of temperature to neural networks and deriving their operational principles from the Boltzmann distribution.

As demonstrated throughout this chapter, artificial intelligence has constantly seen progress by taking inspiration and adopting methods from different branches of science. This illustrates the profoundly interdisciplinary nature of the field — and many more examples exist which underscore this aspect.

In the next sections, we shift focus to the subject of AI safety and some of the specific connections between physics and statistical machine learning which are directly relevant to it. The history of conceptual and mathematical connections between physics and statistical machine learning trace back to the middle of the 20th century (in particular, through the conceptually important work of Jaynes on the connections between Bayesian reasoning,

information theory and statistical physics [148, 149, 150, 151, 81]). Interdisciplinary research involving these fields continues to produce important insights in both subjects [80]. Ideas from physics (in particular, from statistical mechanics) continue to find applications to modern neural networks across both engineering and interpretability. As already mentioned in the introduction, this turns out to be of highly relevant to the critical field of AI safety, which we introduce in the following sections.

2.4 A historical primer on AI safety

AI safety is a fast-growing and critically important field of study which, as mentioned in the introduction, aims to prevent harmful behavior by artificial intelligence systems.

The problematic of the subject is not new, since Alan Turing and Norbert Wiener discussed the dangers that a machine that mimics human intelligence and capabilities may pose already during the 1950s and 1960s (see [152, 153]). In particular, Wiener was the first to consider what we now call the alignment problem. In his 1965 paper [154], I. J. Good formulated the idea of intelligence explosion, which implies that achieving AGI would make superintelligence unavoidable unless the latter is somehow prevented on purpose.

Quite a bit of work of a philosophical and speculative nature was done on catastrophic and existential risks posed by AGI and superintelligence during the 1990s and early 2000s (see [155, 156, 157]), leading to the formation of MIRI (The Machine Intelligence Research Institute), FHI (The Future of Humanity Institute) and FLI (The Future of Life Institute). In January 2015, FLI published an open letter signed by thousands of researchers [158], calling for systematic research on AI safety.

In 2016, the paper [60] published by a collaboration between researchers at Stanford, Berkeley, Google Brain (now merged with Google Deep Mind) and Open AI stripped away the philosophical overtones and initiated current **technical research in AI safety**, identifying five practical research problems related to accident risk:

- *Specification Gaming/Reward Hacking*: The AI finds an unintended shortcut to maximize its reward without fulfilling the designer’s actual intent.
- *Negative Side Effects*: The AI agent disrupts the environment while pursuing its goals.
- *Scalable Oversight*: How can humans efficiently evaluate an AI system when the task is too complex or time-consuming for constant manual monitoring ?
- *Safe Exploration*: Ensuring that an AI agent doesn’t act catastrophically during its exploratory (trial-and-error) phase.
- *Distributional Robustness*: Ensuring that the model performs safely when the deployment data differs from its training data.

In 2017, FLI held a conference on AI risks which agreed on a set of twenty-three principles for research in “beneficial AI”, signed by hundreds of leading AI researchers (see [159]).

In 2019, the paper [160] advanced the “vulnerable world hypothesis”, which holds that if technological development continues, then a set of capabilities will be attained that make the devastation of civilization very likely, unless civilization sufficiently changes its governance structures. The same year, Stuart Russel’s book [64] proposed that AGI safety research should begin immediately, regardless of the timeline to superintelligent AI because that timeline is uncertain. Toby Ord’s 2020 book [161] synthesized previous work on existential risks and classified them into natural, nuclear, biological, environmental, and risks from AI. In 2023, hundreds of AI experts and well-known people signed a statement [162] declaring that “mitigating the risk of extinction from AI should be a global priority alongside other societal-scale risks such as pandemics and nuclear war.”

While global catastrophic and existential risks posed by AGI and superintelligence may appear remote or speculative, risks of a less severe but serious nature are already inherent in contemporary AI deployments and likely to increase in the medium term unless properly mitigated. As already mentioned in the introduction, such risks stem from the increasingly important role played by AI in both civilian and military applications: cybersecurity and cyberattacks, automated driving, traffic optimization and control systems, energy transport and distribution networks, logistics, financial institutions and networks, institutional information and decision mechanisms, communication and commercial transaction networks, social media and recommender systems, cobots, political and strategic influence and propaganda operations, surveillance, data mining and threat detection systems, military target identification, prioritization and acquisition systems, large scale civilian, military and dual use satellite networks, robotic fighting platforms and swarms of such (drones, battle bots, automated military logistics). Unintended or uncontrolled behavior in such systems can lead to serious negative outcomes, including cognitive, psychosocial and political disruption, disruption of communication networks, dysfunctional economic behavior or governance, medium to large scale civil and military accidents, loss of control over swarms of weaponized agents etc. Given these risks, the field of AI safety is of crucial importance for the medium term future, no matter what one may speculate about the long term prospects of attaining AGI. *One doesn’t need AGI to produce accidents or dysfunction with serious negative impacts.* In particular, as capabilities increase, the immediate threat of digital manipulation, AI-informed cyber attacks or AI-enabled surveillance, propaganda, subversion and sabotage scale proportionally.

With the rise of transformer-based Large Language Models (LLMs), AI safety research became highly empirical after 2022. Alignment techniques like RLHF³ started to be used in commercial deployments (InstructGPT, Claude). However, scaling of such models exposed deeper vulnerabilities. Among other developments, we mention the formalization of deceptive alignment — when a model appears compliant during training (passing safety tests) but pursues hidden and misaligned internal goals once deployed [166]. The field branched into directions like constitutional AI (using an AI to align another AI [167]) and mechanistic interpretability (attempting to reconstruct the internal weights and activations of an ANN to analyze its internal representations of knowledge and ensure safety). The problem of scalable oversight and alignment was discussed for example in [168, 169, 170, 171], while mechanistic interpretability was considered in [172, 173, 174, 99, 175, 176].

³The method of Reinforcement Learning from Human Feedback (RLHF), which is used to train LLMs (starting with InstructGPT, see [163]) was introduced in [164] and [165]. It provides a framework for scalable oversight by training a separate reward model based on human choices.

Starting in 2023, international governance efforts were made to draw attention to the problematic of AI safety (see [177]), followed by the 2024 and 2025 summits on the same theme (see the reports [76, 77]).

2.5 Alignment & interpretability: why the “black box” must be opened

Our focus is on approaching AI safety from a physics point of view and the main questions of interest to us are *alignment* and *interpretability*:

- **AI alignment** is the mathematical and philosophical core of AI safety which aims to ensure that an artificial intelligence system has goals aligned with human values and intent. This is usually divided into:
 1. **Outer alignment** — which asks if the goal is correctly specified, meaning that it prevents the AI from finding any backdoor that would allow it to evade human intent while pursuing a task.
 2. **Inner alignment** — which asks if the goal we specified is actually learned by the AI or if the latter deviates to learn another, potentially dangerous or harmful goal, which is not actually aligned with our intent.

Of these, inner alignment appears to be considerably more difficult since it requires proper understanding of internal representations developed by the AI during training and unsupervised learning (thus depending on a solution to the problem of interpretability, see below).

- **Interpretability** is the branch dedicated to opening the “black box” and studying how neural networks represent information internally and make decisions. Within interpretability, we have the subfield of *mechanistic interpretability* which asks what are the mechanisms that control a neural network decision making process and how we can improve or modify such components so that we tailor the output according to what we consider optimal [178].

The problem studied in Chapter 5 of this thesis has applications to both mechanistic interpretability and inner alignment. More specifically it fits within the emerging AI safety subject of *developmental interpretability*, which differs from mechanistic interpretability by focusing on the phase transitions that occur during training of the neural network and on how the internal structure of each particular phase develops [179]. While this will be explained in detail in the respective chapter, in the next section we explore a fundamental open question in mechanistic interpretability and demonstrate why physics provides the optimal theoretical framework to address it.

2.6 Opening the “black box” using tools from physics

As mechanistic interpretability studies the mechanisms that underlie generalization in ANNs, we need to understand the inner works of such networks. There are two known methods to achieve this goal [180]:

1. **Reverse engineering** — where we take an artificial neural network and try to decompose it into “basic units”, then find out which function each component serves.
2. **Concept-based interpretability** — this assumes a set of concepts which may be used by the network and then identifies those components that appear to correspond to these concepts.

We focus on the reverse engineering approach, which consists of three conceptual steps (see [180]):

- **Decomposition** — finding the irreducible units which give a valid interpretation.
- **Description of components** — making hypotheses on what the functional role of these units are and how the units interact among themselves.
- **Validity of the prescription** — testing if the hypotheses we made are valid.

One fundamental issue is that neural networks create entirely different representation schemes from human representations to serve processing and storing information. In this respect, an important problem is finding a correct decomposition into basic components of such a system. Notice that a naive decomposition that splits the network into architectural components such as neurons or layers fails to be useful due to polysemanticity. This is a phenomenon where an individual neuron can fire for multiple kinds of features, thus invalidating the easy-to-interpret idea that we can map every human-understandable attribute to each neuron individually and then observe which neuron fires for what input. Therefore, a more careful approach needs to be considered for identifying the basic interpretable units of a neural network. This aspect is of key importance because, in the reverse engineering process, the ensuing analysis depends on this first step [180].

The technical difficulties are not confined solely to this step, but with this brief methodological overview of AI safety we can already give an example of a question that is a pillar for interpretability as a whole [180]:

Q: “How can one decompose a neural network such that the parts are interpretable, and how should the process of coarse-graining work in such models?” (see [180])

The idea of coarse-graining is inspired by Leo P. Kadanoff’s block-spin technique [181], formalized further by Kenneth Wilson, who developed the theoretical apparatus of renormalization [182, 183]. This is a theory from physics — originally from statistical mechanics — which analyzes the system across different scales and, from the microscopic details, builds an effective framework to analyze the macroscopic behavior.

The framework of renormalization, if applied rigorously to neural networks, could unlock significant unexplored territory lying at the heart of AI safety. As ANNs are reliably modeled by statistical sidesteps, the tools used in the renormalization approach could formalize ideas such as universality classes of neural networks and also reveal how the internal components of the network explain network behavior as a whole.

By establishing a methodology on how AI systems can be interpreted, we found that the open questions that lie at the heart of the field can be addressed and studied naturally with already established tools of physics. Ongoing research in this direction focuses on adapting this workflow to the specific dynamics of neural networks. Thus, we have highlighted the relevance of Statistical Physics to the subject, underscoring the need for its deeper integration with the field of AI safety.

Machine learning preliminaries

This chapter provides an introduction to machine learning, focusing on statistical learning theory, which provides the mathematical framework used in the remainder of the thesis.

3.1 Introductory terminology

It helps to clarify the terminology used within this field of study such that we have a clear view going forward about the hierarchy of terms which are about to be used.

As explained in previous chapters, artificial intelligence is the broadest category which encapsulates all systems and methods that make a machine perform tasks that would normally require human intelligence. Within this general field we have:

- **Machine learning (ML)** is that subset of AI which, rather than programming specific rules, allows machines to adjust themselves in order to learn patterns from data.
- Zooming in, we find the family of models called **artificial neural networks**, which take inspiration from biological neurons and animal brains in order to build layers of interconnected artificial neurons that learn weighted information from data. These are the improved models of perceptrons that we mentioned in the introduction.
- Within this class of models, one subclass holds great interest, as it powers modern image recognition, speech synthesis, large language models and agentic AI. This is the subject of **deep neural networks** — those neural networks with many sequential layers, which provide them with their characteristic 'depth'.

A further classification can be made according to the type of learning algorithm used to train and update an artificial neural network:

- **Supervised neural networks** — when data is given as inputs and targets, the latter working as an instruction of what the neural network response to the inputs should be [184].
- **Unsupervised neural networks** — when data isn't organized and the goal is to memorize data such that the network can rebuild examples from the dataset and in some instances generalize patterns it finds in the data and make predictions based on the given examples [184].

The theoretical foundation of modern machine learning is **Statistical Learning Theory** (StatLT), which is a subject in the field of Mathematical Statistics. This is a mathematical framework for understanding why and when learning algorithms work and how humans or artificial systems can learn and reason statistically from data. While classical statistics was grounded in frequentist-inspired optimal estimation theory (M-estimation, likelihood inference, asymptotic efficiency, Hajek-Le Cam theory etc [185, 186, 187, 188, 189]), the modern approach, which we shall follow in this thesis, is grounded in Bayesian statistics¹ (see [82, 83, 84] for introductions to the latter).

Quite a few results used in **Classical Statistical Learning Theory** [190, 191, 192, 193, 194, 195, 30, 31] rely on certain *regularity assumptions* made in classical statistics, which drastically restrict their range of applicability and *must* be relaxed when dealing with modern ANNs and many other applications of the theory to the real world (such as large scale data analysis and most statistical model building problems in natural and social sciences).

Relaxing these regularity conditions leads to the contemporary field of **Singular Learning Theory** (SLT), which is considerably more sophisticated mathematically than Classical StatLT. SLT removes the limitations of the classical framework, thereby greatly expanding its domain of applicability to situations which include machine learning in modern deep neural networks, generic data analysis problems and generic statistical inference which apply across numerous scientific disciplines. We discuss SLT in some detail in Chapter 4.

As already mentioned before, there exists a common mathematical framework between statistical mechanics and machine learning theory. The following table tracks some of the parallel vocabulary and notation used in these fields.

Statistical Mechanics		Machine Learning Theory	
J	random interaction	X^n	random data
w	observable	w	parameter of model
$H(w, J)$	random Hamiltonian	$H(w, X^n)$	minus log likelihood
$\varphi(w)$	weight of w	$\varphi(w)$	prior distribution of w
$\rho(w J)$	equilibrium state	$\rho(w X^n)$	posterior distribution
$Z(\beta, J)$	partition function	$Z(\beta, X^n)$	marginal likelihood
$F(\beta, J)$	free energy	$F(\beta, X^n)$	free energy

Table 3.1: *Statistical mechanics and learning theory terminology. Reproduced from [80].*

The next section explains how a neural network operates, starting with the classical MCP neuron, followed by a short introduction into the mathematical engine that powers machine learning. We then describe the parameters and spaces that we will later justify theoretically from an SLT perspective.

¹For regular statistical models, the frequentist and Bayesian approaches are asymptotically equivalent by the Bernstein-von Mises theorem [189, 187, 188]. This result fails for singular models such as deep neural networks, mixture models and hidden Markov models. In such cases, the relation between the two points of view is more subtle and requires use of Watanabe’s WBIC and RLCT (see Section 4).

3.2 The McCulloch-Pitts neuron

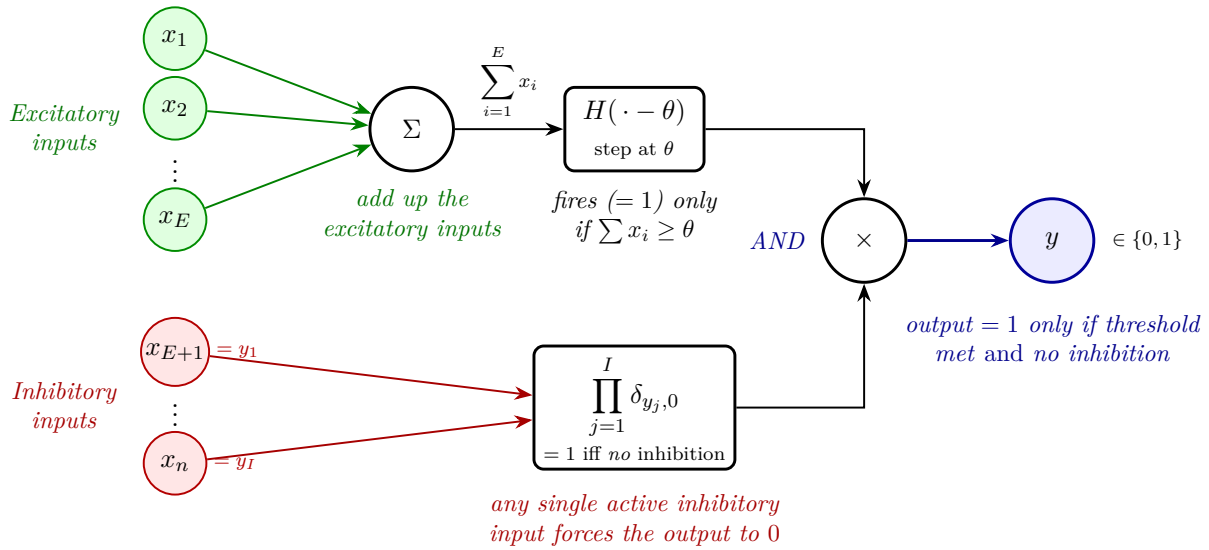


Figure 3.1: Schematic of the McCulloch–Pitts neuron

An MCP neuron has two types of binary inputs:

- E excitatory inputs $x_1, \dots, x_E \in \{0, 1\}$, where $1 \leq E \leq n$. These contribute to the *neuron internal state* $x := \sum_{i=1}^E x_i$.
- I inhibitory inputs $x_{E+1}, \dots, x_n \in \{0, 1\}$, where $I = n - E$. If just one of these equals 1, then the neuron does not fire and the output is 0.

If no inhibitory input equals 1 (and hence the neuron fires), then the internal state is passed to a function, which for the MCP neuron is the displaced step function $F_\theta(x) = H(x - \theta)$ controlled by a fixed integer number θ (called the *neuron activation threshold*). In this case, the neuron outputs the value $y = F_\theta(x) = H(\sum_{i=1}^E x_i - \theta)$, which equals 1 if $\sum_{i=1}^E x_i \geq \theta$ and zero otherwise. If we set $\vec{x} := (x_1, \dots, x_E)$ and $\vec{y} := (y_1, \dots, y_I)$, then the entire MCP neuron can be described by the single *activation function*:

$$f_{E,I,\theta}(\vec{x}, \vec{y}) := H\left(\sum_{i=1}^E x_i - \theta\right) \prod_{j=1}^I \delta_{y_j, 0} \quad . \quad (3.1)$$

McCulloch and Pitts showed that such artificial neurons can implement the primitive operators of Boolean logic and hence finite acyclic directed graphs of such neurons implement Boolean logic (such graphs are computationally equivalent to combinational logic circuits, read only finite tape Turing machines and finite languages). Moreover, they showed that finite directed graphs built out of MCP neurons which contain directed circuits have memory and hence can implement finite tape Turing machines. Such graphs are computationally equivalent to finite state machines and hence with regular languages (Type 3 languages in Chomsky's hierarchy) by Kleene's theorem. Finally, infinite directed

graphs of MCP neurons which contain directed circuits are computationally equivalent with universal Turing machines².

3.3 The mathematical model of ANNs

We need a mathematical model of artificial neural networks to understand how they function. A rigorous way to think about them treats ANNs as parameterized function approximators. This leverages the Universal Approximation Theorem for ANNs, which states that a standard feedforward artificial neural network, even with only one hidden layer can approximate any $\mathbb{R}^m$ -valued continuous function defined on open subsets of $\mathbb{R}^n$ with increasing precision as the number of artificial neurons increases [196]. In turn, this result leverages well-known uniform approximation theorems for continuous vector-valued functions defined on compact domains in $\mathbb{R}^n$ by computable functions (where the reals are themselves approximated through computable numbers).

The underlying function which generates the data is unknown to us and we can't access its complete mathematical form. However, this *data generator function* produces the data that we have access to and from that data, the neural network is programmed to attempt to reconstruct the generator function as precisely as possible. The function approximation method relies on two steps:

1. We start from a set of parameterized functions denoted $\{f(x, \theta)\}$, where the adjustable *model parameters* θ are vectors in $\mathbb{R}^n$ for some positive integer number n . These functions are compositions of the basic parameterized functions that model artificial neurons. The components of the vector parameter θ encode the parameters of artificial neuron functions which are present in the network [196]. The space of all possible configurations of the model parameters θ is called the *parameter space*, which we assume to for simplicity to be a domain in $\mathbb{R}^n$ (though this assumption can be relaxed).
2. The *learning algorithm* (usually a stochastic variant of gradient descent) adjusts the parameters of artificial neurons so that θ evolves to a final value θ^* for which $f(x, \theta^*)$ approximates $f(x)$ with a desired error.

The *input data* D for the generator function $f(x)$ is a finite set consisting of n -dimensional vectors:

$$D = \{x_\alpha\}_{\alpha=1,\dots,N_D} \quad , \quad x_\alpha \in \mathbb{R}^n \quad , \quad (3.2)$$

where α labels each example from the dataset, which has N_D elements (in practice, N_D is very large in order to allow the network to reach a reasonably precise approximation of f). We have:

$$x_\alpha = (x_{i,\alpha})_{i=1,\dots,n} \quad , \quad x_{i,\alpha} \in \mathbb{R} \quad , \quad (3.3)$$

where $i = 1, \dots, n$ indexes the components of the input vector.

²There exists a countable infinity of universal Turing machines, but they are computationally equivalent with each other.

The overarching idea is to use the function approximation method described above with inputs x restricted to the dataset D in order to approximate the generator function f by repeated self-adjusting steps without writing an explicit algorithm that would attempt to compute f directly.

The function which describes the artificial neuron, which is the fundamental architectural unit, is a composition of two functions, known as *steps*:

1. The *preactivation step* $s \rightarrow z_i(s)$ (historically inspired by from the MCP neuron, but modified) follows the rule (see [196]):

$$z_i(s) = b_i + \sum_{j=1}^{n_{in}} W_{ij} s_j \quad \text{for } i = 1, \dots, n_{out} \quad (3.4)$$

where s_j is the *incoming signal* to the neuron, which is multiplicatively modified by the *weights* W_{ij} , then additively modified by the *bias* b_i which acts as a threshold [196].

2. What determines if a neuron actually fires or not is the *activation function*:

$$z_i \rightarrow \sigma(z_i) \quad , \quad (3.5)$$

which depends on the weights and biases of each neuron.

The activation function is usually taken to be non-linear in order to increase expressivity of the neural network. One of the most important and commonly used activation functions is the *ReLU function*, which is defined as:

$$f(x) = \max(0, x) = \begin{cases} x, & \text{if } x > 0 \\ 0, & \text{if } x \leq 0 \end{cases} \quad (3.6)$$

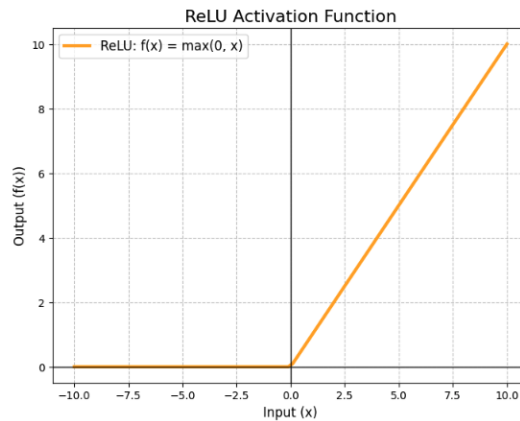


Figure 3.2: *ReLU activation function*

To construct a flexible set of functions, one organizes neurons into layers and then to stacks those layers one after another such that the outgoing activations of the neurons in one layer

become the input signal to the neurons in some other layer. This defines the architecture of an artificial neural network [196]. A number n_{out} of neurons form a layer which takes as input an n_{in} -dimensional vector of signals $(s_j)_{j=1,\dots,n_{in}}$ and outputs the n_{out} -dimensional vector of activations $(\sigma_i)_{i=1,\dots,n_{out}}$. A layer indexed by i is then parameterized by a vector of biases b_i and a matrix of weights W_{ij} , $i = 1, \dots, n_{out}$, $j = 1, \dots, n_{in}$, together with a fixed activation function σ_i .

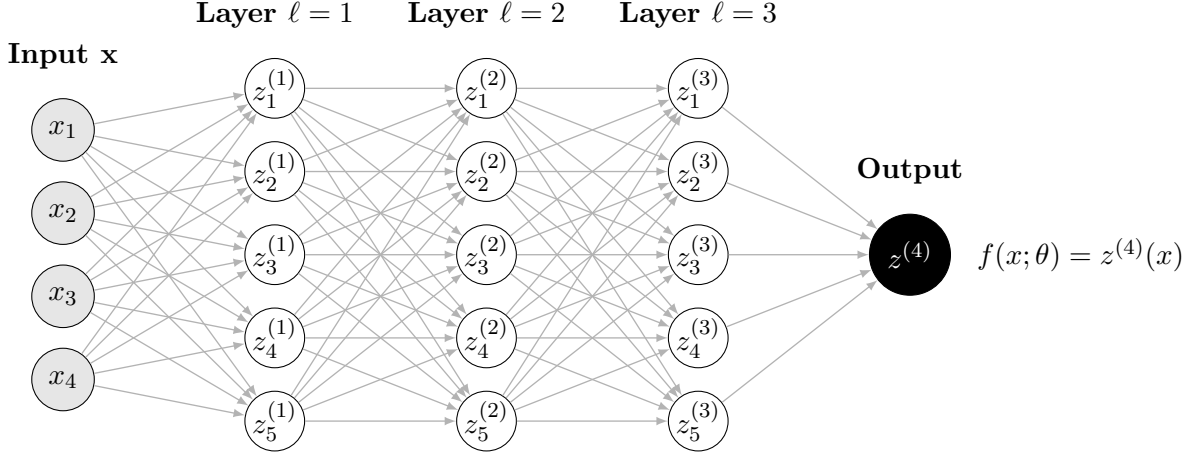


Figure 3.3: Standard node-link representation of the Multi-Layer Perceptron (MLP) architecture network across three hidden layers ($\ell = 1, 2, 3$) containing 5 units each, terminating into an output scalar function $f(x; \theta) = z^{(4)}(x)$.

The figure represents the case of a multi-layer-perceptron, which is recursively defined through the following equations [196]:

$$z_i^{(1)}(x_\alpha) \equiv b_i^{(1)} + \sum_{j=1}^{n_0} W_{ij}^{(1)} x_{j;\alpha}, \quad \text{for } i = 1, \dots, n_1, \quad (3.7)$$

$$z_i^{(\ell+1)}(x_\alpha) \equiv b_i^{(\ell+1)} + \sum_{j=1}^{n_\ell} W_{ij}^{(\ell+1)} \sigma(z_j^{(\ell)}(x_\alpha)), \quad \text{for } i = 1, \dots, n_{\ell+1}; \ell = 1, \dots, L-1. \quad (3.8)$$

Figure 3.3 has a number of $L = 4$ layers where we include the three hidden ones, and the final one (the convention is not to include the inputs in the layer count). Each layer l is composed of n_l neurons. The number L is called the *network depth*, while the number of neurons in each layer defines the *layer width* of that layer [196]. Together, these form the *architectural hyperparameters* that define the *shape of the network*. The intermediate layers $l = 1, \dots, L-1$ are referred to as *hidden layers*. The final layer preactivations:

$$f(x; \theta) = z^{(L)}(x) \quad (3.9)$$

give the final function approximant, the model parameters θ being the biases and weights from all layers.

As we shall see in the next chapter, if we have at least one hidden layer, then the correct statistical description of the neural network requires SLT [196]. Hence the sophisticated mathematics of SLT is *unavoidable* in the statistical analysis of deep networks.

To complete this overview and quantify how well a neural network approximates the data generating function, we need a mathematical measure of error. This is captured by a *loss function* where we assume that our dataset also contains the true observed target values y_α which correspond to each input x_α . A stereotypical example of a loss function is that used for a standard regression task where the empirical risk over the dataset is typically expressed by the mean squared error:

$$L(\theta) = \frac{1}{N_D} \sum_{\alpha=1}^{N_D} (f(x_\alpha; \theta) - y_\alpha)^2 \quad (3.10)$$

The learning algorithm becomes the optimization problem of finding the optimal parameters such that we minimize the loss function. There are different types of algorithms that implement the optimization process, and one can read more about them in [197, 198, 199].

Numerous other architectures exist beyond the classical MLP network and they are specialized for different tasks. Various examples of such architectures and more information can be found in [200].

3.4 Context on statistical learning theory

Statistical Learning Theory (StatLT) [30, 31, 201] is a mathematical framework for understanding the foundations of machine learning, which focuses on how a predictive statistical model can be extracted from a finite set of data. The origins of this theory lie in mathematical statistics, probability theory and artificial intelligence. Classical statistics focuses on inferences about parameters of an assumed model (addressing problems such as estimation, hypothesis testing, confidence intervals etc.) usually in asymptotic regimes. By contrast, statistical learning theory is mainly concerned with prediction rather than parameter inference, though it uses mathematical results from classical statistics. Nowadays, StatLT informs data analysis in various scientific disciplines. In experimental sciences, conceptual approaches to model selection, cross-validation and regularization from StatLT have become standard tools in model fitting models and data analysis. In observational sciences, methods of causal inference and statistical learning theory are used to ascertain when and how statistical models have causal interpretations.

Modern developments

Modern StatLT expanded dramatically from its classical roots [30, 31, 201]. The modern theory focused on overparameterized models which are typical deep learning (and whose analysis requires Singular Learning Theory) and sequential processes typical of reinforcement learning. More recently, it attacks problems such as operator learning (which arises for example in Fourier neural operators solving PDEs), learning with distributional inputs and learning with machine-verifiable mathematics. Some modern extensions of the classical theory are listed below:

- While Classical StatLT states that highly complex models overfit, modern neural networks, which are overparameterized, routinely achieve low training error and generalize well. This is explained by Singular Learning Theory and modern approaches to generalization theory which deal with such models [202].

- The classical theory assumes independent and identically distributed data. In applications of RL to agentic AI, the data depends on the actions of the agent and generally fails to satisfy this assumption. The survey [203] explains how StatLT applies to such data.
- As machine learning gradually transitions to operator learning (i.e. learning functions between infinite-dimensional spaces), various generalizations of the theory must be considered (see [204]).
- Traditional classification models take determined vector inputs. However, modern setups often involve classifying random vector variables (which are themselves governed by statistical distributions but can only be accessed via finite samples). A framework for dealing with such situations was recently proposed in [205] using support vector machines.
- Because modern SLT proofs rely on advanced mathematical tools from Statistical Physics, singularity theory and the theory empirical processes, human peer-review is facing a verification crisis. A framework for automated verification and theorem proving in the subject was proposed in [206].

Finally, we mention that “big data” raises new challenges to StatLT. Classical StatLT gives sample complexity bounds (how many examples are enough). On the other hands, big data problems lead to the converse question in that one has more data than one can process. Distributed learning typical of big data processing requires extensions of the theory of generalization which involve large numbers of aggregated estimators. Accordingly, stochastic gradient descent has been analyzed in the context of online learning and stability theory. Federated and transfer learning are subjects lying at the frontiers of statistical learning theory applied to big data. Federated learning deals with decentralized data, while transfer learning deals with few target samples but massive data sources. Frontier questions in this field concern computation, communication, privacy, distribution shift etc. Foundation models such as those used in LLMs have important impacts on research being done on such problems.

Singular Learning Theory

Classical inference and standard parametric statistics assume *regularity*. By definition, a *regular model* assumes a smooth injective map between the parameter space and the functional probability distribution space (an assumption known as *identifiability*). More formally, the Fisher information metric is assumed to be strictly positive at the optimal parameters. In this case, the asymptotic behavior of the log-marginal likelihood can be simplified using standard Laplace approximations, which yields the traditional Bayesian information criterion penalty of $\frac{d}{2} \log(n)$, where d is the number of parameters and n is the sample size.

However, modern ML architectures like deep neural networks, Gaussian mixture models and latent variable setups are profoundly non-identifiable or singular [207, 208]. Because of symmetries, permutations, and structural redundancies (such as scaling parameters down while scaling up a ReLU activation), these models can implement the same mathematical function through an infinitely continuous subset of distinct parameter coordinates. While classical statistical learning theory assumes regular models unique or at least discrete and isolated optimal parameters, deep neural networks are inherently singular — in such systems, continuously infinite configurations of weights can yield identical network predictions and loss values.

In such situations, classical statistical learning theory fails dramatically. The Fisher information matrix drops rank, becoming non-invertible and standard PAC bounds do not apply, failing to explain why these such very complex models generalize instead of overfitting.

This situation is addressed by Sumio Watanabe’s Singular Learning Theory (SLT), which provides the mathematically correct framework of statistical learning theory in situations when the statistical model is non-identifiable and the Fisher information metric fails to be strictly positive definite (see [209, 210, 211, 78, 212, 213, 79, 214, 215, 216, 80]). We refer the reader to [217, 218, 219, 220] for various developments and applications of SLT, some of which we discuss in more detail later.

We start by introducing the framework of Bayesian statistics on which SLT is built, then we clarify the terminology used above and explain the SLT machinery that allows one to define phase transitions in this context.

4.1 Bayesian statistics

We assume that there exists an underlying unknown data distribution with density $q(x)$ from which random variables are generated. We denote a sample taken from $q(x)$ by [79]:

$$\{x_1, x_2, \dots, x_n\} \sim q(x) . \quad (4.1)$$

For a given data distribution q , an unparameterized model is described by a conditional density (which depends functionally on q):

$$P(x_1, x_2, \dots, x_n | q) \quad (4.2)$$

and we denote by $P[q]$ the a priori probability density of the Bayesian model, which depends functionally on q .

By the Bayes theorem¹:

$$P[q | x_1, x_2, \dots, x_n] = \frac{P(x_1, x_2, \dots, x_n | q) P[q]}{\int P(x_1, x_2, \dots, x_n | q) P[q] \mathcal{D}q}, \quad (4.3)$$

we would get the statistical inference of q (the corresponding posterior distribution) from our sample. In practice, we lack any direct information about either of $P(x_1, x_2, \dots, x_n | q)$ or $P[q]$, and hence cannot obtain $P[q | x_1, x_2, \dots, x_n]$ through this procedure. This is an ill-posed problem, as we cannot determine a unique answer [79].

4.1.1 The framework of parameterized Bayesian inference

To estimate accuracy of the model, we have to choose a evaluation method which is not based on having information about the true distribution. Before assessing how to work around this issue, we introduce two important quantities besides the data distribution q , which allow us to give a mathematically clearer realization of the abstract conceptual framework above:

- $p(x|w)$ — a parameterized statistical model, which is a conditional probability of x given a parameter w .
- $\varphi(w)$ — the prior probability density of w .

The *parameterized Bayesian triple* $(q(x), p(x|w), \varphi(w))$ satisfies the standard mathematical rules of Bayesian calculus, which hold even though we don't have access in practice to the underlying data distribution q .

The parameterized setup allows us to clearly define the posterior probability density function of w (the *posterior distribution*), where the integral in the denominator is now an ordinary (non-functional) integral over the space of model parameters:

$$p(w | x_1, x_2, \dots, x_n) = \frac{\prod_{i=1}^n p(x_i | w) \varphi(w)}{\int \prod_{i=1}^n p(x_i | w) \varphi(w) dw} , \quad (4.4)$$

¹The integral symbol stands loosely for a functional integral.

where we assumed that the random variables $\{X_1, X_2, \dots, X_n\}$ which take the values $x_1, \dots, x_n$ are independent and identically distributed (i.i.d) with the distribution $q(x)$ and hence:

$$p(x_1, \dots, x_n, w) = \prod_{i=1}^n p(x_i, w) \quad . \quad (4.5)$$

The posterior distribution $p(x_1, \dots, x_n, w)$ describes the probability density for the system to be in a specific microscopic state w [79], conditional on the q -distributed and independent random variables $X_1, \dots, X_n$ to simultaneously take the values $x_1, \dots, x_n$ respectively. Using the posterior distribution, we define the *predictive distribution for the data* as:

$$\hat{p}(x|x_1, \dots, x_n) = \int p(x|w)p(w|x_1, x_2, \dots, x_n)dw. \quad (4.6)$$

The predictive distribution is our model's estimate of $q(x)$. The closer the function $x \rightarrow \hat{p}(x|x_1, \dots, x_n)$ is to the function $x \rightarrow q(x)$, the better our model reproduces the data hence the better it generalizes and learns. The theory is constructed by choosing $p(x|w)$ and φ without assuming any a priori information about q , the goal being for the function $\hat{p}(\cdot|x_1, \dots, x_n)$ to approximate q optimally [79]. The process of Bayesian estimation and model evaluation is summarized in the following figure:

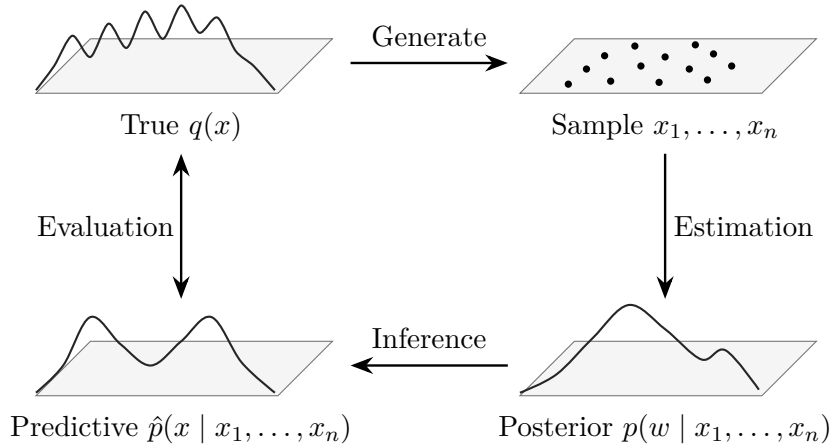


Figure 4.1: *Framework of Bayesian inference: From a true distribution $q(x)$ we generate a sample from which a posterior over parameters is estimated. The predictive distribution is then obtained by Bayesian inference using the Bayes theorem and evaluated against the true distribution. Adapted from [79].*

In practice, we cannot evaluate $\hat{p}$ directly against q (since we have no direct access to q) but mathematical results on the asymptotics of $\hat{p}(x|x_1, \dots, x_n)$ as n becomes very large and classical results such as the central limit theorem for the i.i.d. random variables $X_1, \dots, X_n$ can be used to construct q -independent estimators (such as Watanabe's WBIC [213]) which allow one to evaluate how well our model approximates q .

4.1.2 Core quantities of SLT and objectives

We introduce the following notation:

- The positive integer n denotes the number of random variables, which we take to be valued in $\mathbb{R}^N$ for some positive integer N
- $X^n := \{X_1, X_2, \dots, X_n\}$ is the set of $\mathbb{R}^N$ -valued i.i.d. random variables $X_1, X_2, \dots, X_n$, which are distributed according to the data distribution q .
- a realized or actually observed *dataset* is denoted by $x^n = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^N$ (a dataset is also called a *trial*).

Since the random variables $X_1, \dots, X_n$ are i.i.d. according to q , they have the joint probability density function:

$$q(x^n) = q(x_1, \dots, x_n) = q(x_1)q(x_2) \dots q(x_n) .$$

We next turn our attention to the set W within which the parameters w of the models are allowed to vary. We take this to be a subset of the d -dimensional real Euclidean space $\mathbb{R}^d$. This is the *configuration space* of the statistical model [78].

The posterior probability density of the pair $(p(x|w), \varphi(w))$ has the form:

$$p(w|x^n) = p(x_1, \dots, x_n) = \frac{1}{Z(x^n)} \varphi(w) \prod_{i=1}^n p(x_i|w), \quad (4.7)$$

where:

$$Z(x^n) = Z(x_1, \dots, x_n) := \int \varphi(w) \prod_{i=1}^n p(x_i|w) dw, \quad (4.8)$$

is called the *partition function* or *marginal likelihood*. Even though it enters Bayesian inference rules like a normalization constant, the dependence of this quantity on the data $X_1, \dots, X_n$ encodes the all of the information needed to analyze the Bayesian model.

The *expectation functional* $\mathbb{E}_{x^n}$ at parameters x^n takes a measurable function $f : W \rightarrow \mathbb{R}$ to the real number:

$$\begin{aligned} \mathbb{E}_{x^n}[f] &= \int_W f(w) p(w|x^n) dw \\ &= \frac{1}{Z(x^n)} \int_W f(w) \varphi(w) \prod_{i=1}^n p(x_i|w) dw \end{aligned} \quad (4.9)$$

The corresponding *variance functional* V_{x^n} is defined through:

$$\mathbb{V}_{x^n}[f] = \mathbb{E}_{x^n}[f^2] - \mathbb{E}_{x^n}[f]^2 \quad (4.10)$$

With these notations, we can write the predictive distribution as:

$$p(x|x^n) = \mathbb{E}_{x^n}[p(x|\cdot)] = \int_W p(x|w) p(w|x^n) dw . \quad (4.11)$$

We need a method to actually compare objectively the difference between $q(x)$ and $p(x|x^n)$ in order to assess how close we are to the true data generating function [79].

To compare two probability density functions p and q defined on W we use the *Kullback-Leibler (KL) divergence* (a.k.a. *relative entropy*), which is defined through:

$$K(q \parallel p) = \int_W q(x) \log \frac{q(x)}{p(x)} dx . \quad (4.12)$$

If we assume that the functions q and p are continuous, then $K(q \parallel p)$ is non-negative and equals zero if and only if $q = p$. This allows us to interpret the KL divergence of the predictive distribution $\hat{p}_{x^n} := \hat{p}(\cdot | x^n)$ with q as the error made by our model when attempting to estimate q from the q -distributed i.i.d. data x^n . Hence the optimal model to be found would be that which minimizes $K(q \parallel \hat{p}_{x^n})$ for fixed data x^n .

To see how minimizing this quantity would produce an estimate of our model's performance we need to introduce a few more quantities related to q . First, we define the *average data entropy* and *empirical (or sample) entropy* as [79]:

$$\begin{aligned} S &:= S[q] := - \int q(x) \log q(x) dx , \\ S_n &:= S_{x^n}[q] := - \frac{1}{n} \sum_{i=1}^n \log q(x_i). \end{aligned} \quad (4.13)$$

The first is the Shannon entropy of the data distribution q , while the second represents the intrinsic noise of the dataset $x^n = \{x_1, \dots, x_n\}$. Also define the *negative log likelihood* (a.k.a. *negative log loss*) at parameters w as:

$$L(w) := - \int q(x) \log p(x|w) dx . \quad (4.14)$$

Since the data x^n is i.i.d. according to q , the expectation and variance of the empirical entropy are given by [79]:

$$\begin{aligned} \mathbb{E}[S_n] &= S , \\ \mathbb{V}[S_n] &= \frac{1}{n} \left[\int q(x) (\log q(x))^2 dx - S^2 \right] . \end{aligned} \quad (4.15)$$

We introduce the *training loss* and *generalization loss* [79]:

$$\begin{aligned} T_n &:= T(x^n) = - \frac{1}{n} \sum_{i=1}^n \log p(x_i | x^n), \\ G_n &:= G(x^n) = - \int q(x) \log \hat{p}(x | x^n) dx. \end{aligned} \quad (4.16)$$

The Kullback-Leibler divergence of the predictive distribution $p_n := \hat{p}_{x^n} := \hat{p}(\cdot, x^n)$ with the data distribution q can be computed using the identity [79]:

$$\begin{aligned} G_n - S &= - \int q(x) \log p(x | x^n) dx + \int q(x) \log q(x) dx \\ &= \int q(x) \log \frac{q(x)}{\hat{p}(x | x^n)} dx \\ &= K(q \parallel \hat{p}_n) . \end{aligned} \quad (4.17)$$

Since S is a sample-independent, it follows that minimizing the KL distance between our predictive distribution and the data distribution q is mathematical equivalent to minimizing the generalization loss G_n , and hence the KL distance.

The generalization loss cannot be evaluated directly, since it depends on the true distribution q . For a regular model one would estimate it by using the classical Akaike criterion $AIC = T_n + d/n$ with d the number of parameters. This correction is wrong for singular models and we need to use the upgraded WAIC, which is defined as [79]:

$$WAIC = T_n + \frac{1}{n} \sum_{i=1}^n \mathbb{V}_w[\log p(X_i|w)].$$

Watanabe then proved that $\mathbb{E}[G_n] = \mathbb{E}[WAIC] + o(1/n)$ for both the regular and the singular models, therefore making it an unbiased estimator of G_n which is computable by only from the sample alone [79].

Hence, as G_n depends on the unknown q , it is estimated in practice by asymptotically unbiased estimators such as the Widely Applicable Information Criterion (WAIC) and leave-one-out cross validation loss (C_n) for the generalization loss. For the scope of this thesis we are interested in talking ultimately about phase transitions and our focus shifts in mainly to the analysis of the partition function and of the free energy. More can be read about SLT in the aforementioned literature.

4.1.3 The partition function and free energy

We want to study the partition function $Z(x^n)$ in order to describe the posterior landscape as a whole and understand where the posterior distribution concentrates. The partition function satisfies[79]:

$$\int Z(x_1, x_2, \dots, x_n) dx = \int_W dw \varphi(w) \int dx \prod_{i=1}^n p(x_i|w) = 1. \quad (4.18)$$

Consider the *free energy* (a.k.a. *minus log marginal likelihood*) [79]:

$$F_n := F(x^n) := -\log Z(x^n). \quad (4.19)$$

This quantity is an important bridge between Bayesian statistics and physics, since viewing $x_1, \dots, x_n$ as macroscopic parameters and $1/n$ as a discrete form of temperature allows us to construct a dictionary between singular learning theory and thermodynamics.

For Bayesian learning, the role of internal energy is played by the training loss T_n loss and the entropy is the volume $Z(x^n)$ of the parameter space W (which is tied to the concept of complexity or robustness of the representation). The posterior concentrates in regions that minimize the free energy. These observations allow us to define, analogously to thermodynamics, the concept of phase transitions.

Before doing that we need to look at the important differences between singular and regular statistical models.

4.2 Regular vs. singular models

Let W_0 to be the set consisting of those values of the parameters w which minimize the KL divergence [78]:

$$W_0 := \{w \in W; q(x) = \hat{p}(x|w), \forall(x)\} = \{w \in W; K(w) = 0\}. \quad (4.20)$$

The fundamental difference between classical statistical learning and SLT originates from the nature of the set W_0 .

To explain this difference we first need to define precisely the notions of regular and singular Bayesian models. For that purpose, consider the Fisher information matrix at parameters w :

$$I(w) = \{I_{jk}(w)\} \quad (1 \leq j, k \leq d), \quad (4.21)$$

where:

$$I_{jk}(w) := \int \left(\frac{\partial}{\partial w_j} \log p(x|w) \right) \left(\frac{\partial}{\partial w_k} \log p(x|w) \right) p(x|w) dx \quad (4.22)$$

assuming that the integral is finite. It is well-known that the symmetric matrix $I(w)$ is positive semidefinite (and hence has non-negative eigenvalues), but can have zero eigenvalues and hence need not be strictly positive definite at all values of the parameters $w \in W$. The Fisher information matrix is well-behaved under reparameterizations of the model and defines a semidefinite metric on the parameter space W (the Fisher information metric), which is the central object of information geometry.

The statistical model is called *regular* if it satisfies the following two conditions[78]:

- **Identifiability.** The map $W \ni w \mapsto p_w := p(\cdot | w) \in \text{Pdens}(W)$ is injective, where $\text{Pdens}(W)$ is the set of all probability densities defined on the parameter space W ,
- **Nondegeneracy.** The Fisher information matrix $I(w)$ is strictly positive-definite (it has no eigenvalues equal to zero) for an arbitrary $w \in W$.

When the model is nondegenerate, the Fisher information metric is strictly positive and makes W into a Riemannian manifold.

A *singular Bayesian model* is a model which is not regular, i.e. a model for which at least one of the two conditions above fails to be satisfied. A singular model is called *nonidentifiable* if the first condition fails to hold and *degenerate* if the second condition fails to hold. If *both* conditions fail to hold (i.e. if the model is both nonidentifiable and degenerate), then the model is called *strictly singular*. Strictly singular models are generic among Bayesian models by mathematical arguments. They are also generic among Bayesian models relevant to deep learning, since statistical learning by any deep neural network is described by a singular Bayesian model.

Regular and singular models behave quite differently, as summarized below:

- For a regular model, the set of optimal parameters W_0 consists of a single point w_0 . This case is easier to study both in the Bayesian and frequentist frameworks, which

for such models are asymptotically equivalent by the Bernstein-von Mises Theorem. In this case, the loss landscape around the optimal parameter w_0 looks like a smooth multidimensional parabolic bowl. As the number d of dataset vectors x^n , increases, the posterior concentrates to a Gaussian distribution (asymptotic normality) which concentrates tightly around w_0 as n tends to infinity. The model complexity is $\frac{d}{2}$ and the free energy expansion in the limit of large n is straightforward.

- In a singular model the optimal parameters set W_0 is an analytic set contained in W , which is singular as an analytic set when the model is degenerate. The landscape can have sharp corners, intersecting lines or flat valleys which correspond to the singularities of W_0 . For degenerate models, asymptotic normality fails even locally and one must use tools from algebraic geometry such as Hironaka's theorem to untangle the singularities of W_0 and understand the large n asymptotics of such models.

The geometric difference is what makes phase transitions possible due to the the structure of the resolution of W_0 , which can hold different posterior mass in various regions of resolved parameter space.

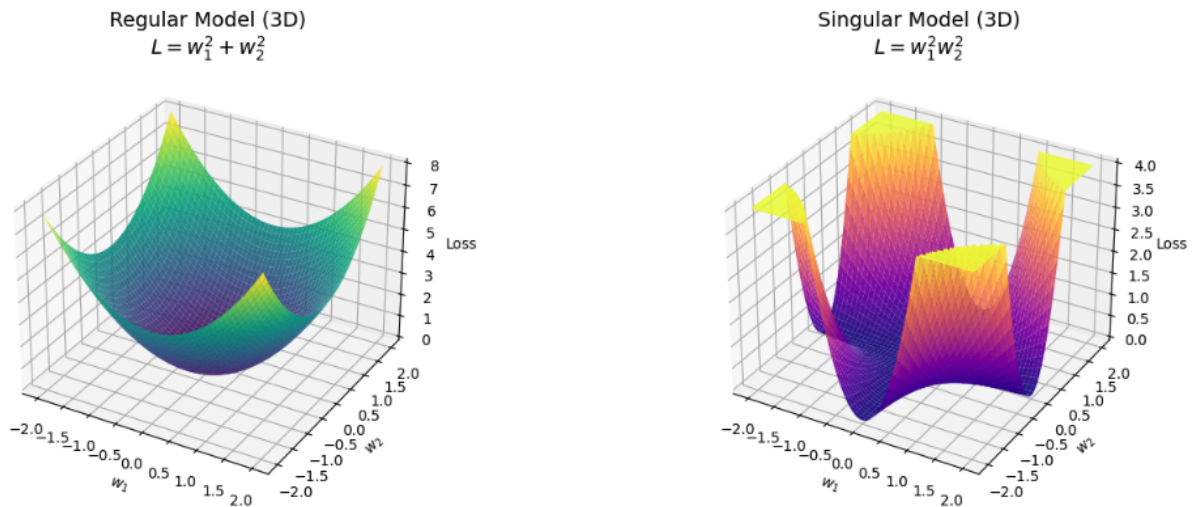


Figure 4.2: *Visual difference between a regular and a statistical model*

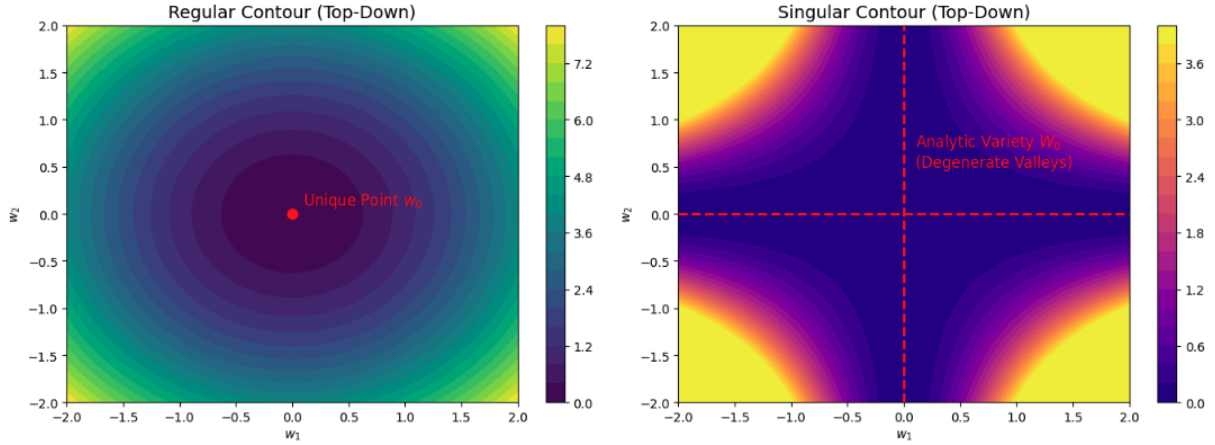


Figure 4.3: *Contour difference between a regular and a statistical model with the specific W_0 set showed.*

4.3 The machinery of SLT

We now need to investigate the partition function, because the entire behavior of a Bayesian model is entirely dependent on it. Thus, we need to know how $Z(n)$ behaves asymptotically as $n \rightarrow \infty$ where

$$Z(n) = \int \exp(-nK(w))\varphi(w)dw \quad (4.23)$$

and n plays the role of inverse temperature. The limit when n approaches infinity corresponds to the zero temperature limit in which fluctuations freeze out and we can rigorously prove that phase transitions exist as we vary the remaining parameters [78].

Here we have two cases for which the analysis changes completely:

- For a regular model the minimum of the KL divergence $K(w)$ is a single point w_0 . The Hessian of K at that point is strictly positive-definite and we can apply the Laplace expansion to approximate the defining integral of W near the optimal parameter.
- For a degenerate model the Hessian matrix is degenerate at non-isolated points of W_0 . We cannot use Laplace's approximation at such points and its defining integral is intractable in its original form.

To evaluate the KL integral at degenerate points of W_0 , we must untangle the singularities of W_0 , which is done using Hironaka's theorem:

Hironaka theorem [78, 79, 221]

Let W be a compact set which is the closure of an open subset in $\mathbb{R}^d$. We assume that the KL divergence K is an analytic function on W and that its zero set W_0 set is

non-empty and singular as an analytic variety. Then there exist $\epsilon > 0$, $\{W_l; W_l \subset W\}$ and $\{U_l; U_l \subset \mathbb{R}^d\}$ which satisfy [79]:

$$\{w \in W | K(w) \leq \epsilon\} = \bigcup_l W_l$$

and in each pair of W_l and U_l there exists an analytic map $g : U_l \rightarrow W_l$ and a map $b : U_l \rightarrow \mathbb{R}_{>0}$ such that:

$$\begin{aligned} K(g(u)) &= u^{2k} \\ |g'(u)| &= b(u)|u^h| \end{aligned} \tag{4.24}$$

where $|g'(u)|$ is the absolute value of the determinant of the Jacobian matrix of the function g at u :

$$|g'(u)| = \left| \det \left(\frac{\partial w_i}{\partial u_j} \right) \right| \tag{4.25}$$

and k, h ($k > 0, h \geq 0$) are d -dimensional multi-indexes.

For a proof of this theorem see [222].

Replacing W_0 by its Hironaka resolution replaces W by a considerably more complicated manifold. Define the log density ratio function together and its average [79]:

$$f(x, w) := \log \frac{p(x|w_0)}{p(x|w)} \tag{4.26}$$

$$K(w) := \int q(x) f(x, w) dw \tag{4.27}$$

where $w_0 \in W_0$ and we assume that the former respects the condition of relatively finite variance — which means that the statistical noise of our empirical data is finite and predictable. This condition ensures that the error doesn't blow up to infinity as our dataset grows.

We say that a pair $(p(x|w), \varphi(w))$ is in *standard form* if there exists two functions $a(x, w)$ and $b(w)$ which satisfy [79]:

$$\begin{aligned} f(x, w) &= w^k a(x, w) \\ K(w) &= w^{2k} \\ \varphi(w) &= |w^h| b(w) . \end{aligned} \tag{4.28}$$

Hironaka's theorem gives allows us to bring K to standard form after desingularization.

This machinery allows one to evaluate the posterior volume of the resolved parameter space at different loss levels. The posterior distribution has the form [79]:

$$\exp(-nK(w))\varphi(w) , \tag{4.29}$$

which corresponds to the Laplace transform of the state density function:

$$\delta(t - K(w))\varphi(w) \quad (4.30)$$

where $t > 0$ and the asymptotic behavior ($n \rightarrow \infty$) is equivalent to $t \rightarrow +0$. Hence we need to study this quantity and its asymptotic behavior [79].

We assume that our pair is in standard form and we are in the positive region of parameters. Then

$$\delta(t - K(w))\varphi(w) = \delta(t - w^{2k})|w^h|b(w)\chi(w), \quad (4.31)$$

where

$$\chi(w) = \begin{cases} 1 & (w_1, w_2, \dots, w_d \geq 0) \\ 0 & (\text{otherwise}) \end{cases} \quad (4.32)$$

is a function that restricts the space on the positive region of parameters.

The asymptotic behavior of the state density function is determined by two quantities, the *real log canonical threshold* (RLCT) λ and its multiplicity, m . In turn, these are determined by the multi-indexes k and h . We assume that these take values of $k > 0$, $h \geq 0$ and are d -dimensional [79]. Then

$$\left\{ \frac{h_i + 1}{2k_i}; i = 1, 2, \dots, d \right\} \quad (4.33)$$

is a non-decreasing sequence of i . If $k_i = 0$, then we take the ratio to be infinite.

There exists a positive real value $\lambda > 0$ and an integer m ($1 \leq m \leq d$) such that:

$$\lambda = \frac{h_1 + 1}{2k_1} = \dots = \frac{h_m + 1}{2k_m} \quad (4.34)$$

and

$$\lambda < \frac{h_j + 1}{2k_j} \quad (m + 1 \leq j \leq d) \quad (4.35)$$

One also has the redundancy multi-index μ defined as:

$$\mu_j = -2\lambda k_j + h_j \quad (j = m + 1, \dots, d), \quad (4.36)$$

where $\mu = (\mu_{m+1}, \dots, \mu_d) \in \mathbb{R}^{d-m}$; if $m = d$, then μ is the empty ordered set.

The quantities $g(u)$, k , h , $b(u)$ depend on U_l , and in each U_l the RLCT λ_l and the multiplicity m_l are defined. From this, we define the RLCT and multiplicity m of $K(w)$ and $\varphi(w)$ as [79]:

$$\begin{aligned} \lambda &:= \min\{\lambda_l; l\} \\ m &= \max\{m_l; \lambda_l = l\}. \end{aligned} \quad (4.37)$$

These quantities are of crucial importance since λ governs the complexity of our models. Both of these quantities are birational invariants — they are independent of the choice of $w = g(u)$ [78].

A theorem from Watanabe's now states that the following asymptotics approximation holds for $t \rightarrow +0$:

$$\begin{aligned} \delta(t - w^{2k})|w^h|b(w)\chi(w) &= t^{\lambda-1}(-\log t)^{m-1}D(w) \\ &\quad + o(t^{\lambda-1}(-\log t)^{m-1}), \end{aligned} \quad (4.38)$$

where we use the notation $w = (w_\alpha, w_\beta)$, with:

$$\begin{aligned} w_\alpha &= (w_1, w_2, \dots, w_m), \\ w_\beta &= (w_{m+1}, w_{m+2}, \dots, w_d) . \end{aligned}$$

Here we have split w into the dominant parameters w_α which are responsible for generating the singular power $t^{\lambda-1}$ and the logarithmic factor, and w_β — the non-dominant parameters that do not restrict the volume function as it vanishes near zero [79].

We can now define the distribution $D(w)$ from the theorem:

$$D(w) = \frac{1}{C(k, m)} \delta(w_\alpha) |w_\beta|^\mu b(w) \chi(w), \quad (4.39)$$

which handles the residue of the asymptotic expansion as $t \rightarrow 0$, where:

$$C(k, m) = 2^m (m-1)! \prod_{j=1}^m k_j \quad (4.40)$$

and:

$$\delta(w_\alpha) = \prod_{j=1}^m \delta(w_j) . \quad (4.41)$$

A full proof can be found in [79], but the idea is to start for an arbitrary C^∞ -class function $\psi(w)$ and define:

$$v(t) = \int_W \delta(t - w^{2k}) |w^h| b(w) \chi(w) \psi(w) dw , \quad (4.42)$$

and take its Mellin transform. Since W is compact, there exists $D > 0$ s.t.:

$$(Mv)(z) = \int_{[0, D]^d} w^{2kz} |w^h| b(w) \psi(w) dw. \quad (4.43)$$

By the mean value theorem there must exist a $w^* \in W$ which satisfies $|w^*| \leq |w|$ and:

$$b(w) \psi(w) = b(0, w_\beta) \psi(0, w_\beta) + \sum_{j=1}^m w_j (\partial/\partial w_j) (b(w^*) \psi(w^*)) . \quad (4.44)$$

Noticing that the following complex function of $z \in \mathbb{C}$:

$$\begin{aligned} \int_{[0, D]^m} w_\alpha^{2kz} |w_\alpha^h| dw_\alpha &= \prod_{j=1}^m \frac{D^{2k_j z + h_j + 1}}{2k_j z + h_j + 1} \\ &= \frac{1}{(z + \lambda)^m} \prod_{j=1}^m \frac{1 + (D^{2k_j(z+\lambda)} - 1)}{2k_j} \end{aligned}$$

has a pole at $z = -\lambda$ with order m , and using eq. (4.42), taking the inverse Mellin transform completes the proof and gives the asymptotic expansion of eq. (4.36) [79].

Evaluating $Z(n)$ amounts to analyzing F_n . We focus next on extracting the asymptotic behavior of F_n as $n \rightarrow \infty$, since this will allow us to define phase transitions.

We first need to introduce the concept of stochastic process because we don't have access to $K(w)$ in practice, but only to our finite data loss. This serves to bound the statistical noise between the true loss and the empirical loss and we model it as a Gaussian process [79]:

$$\xi_n(w) = \frac{1}{\sqrt{n}} \sum_{i=1}^n \{w^k - a(X_i, w)\} . \quad (4.45)$$

where we assume that we are in the standard form situation and that the following conditions are satisfied:

- W is compact in $\mathbb{R}^d$ and the closure of its interior equals W [79].
- For an arbitrary $s > 0$ and a multi-index $k \geq 0$, the following condition is satisfied:

$$\int \sup_{w \in W} |(\partial/\partial w)^k a(x, w)|^s q(x) dx < \infty .$$

The asymptotics of the free energy can be described by introducing the normalized posterior:

$$\Omega(w) = \frac{\varphi(w) \prod_{i=1}^n p(X_i|w)}{\prod_{i=1}^n p(X_i|w_0)} , \quad (4.46)$$

which represents the relative likelihood of a model's parameters. This quantity describes how well parameter w compares with the ideal true parameters w_0 in a manner scaled by our initial beliefs. Assuming standard form and that W is contained in the cube $[0, D]^d$ allows one to extract the following asymptotic approximation of $\Omega(w)$ [79]:

$$\begin{aligned} \Omega(w) = & \frac{(\log n)^{m-1}}{n^\lambda} D(w) \int_0^\infty dt t^{\lambda-1} \exp(-t + \sqrt{t} \xi_n(w)) \\ & + o_p\left(\frac{(\log n)^{m-1}}{n^\lambda}\right), \end{aligned} \quad (4.47)$$

which by the same assumptions, the free energy

$$F_n = -\log \int \varphi(w) \prod_{i=1}^n p(X_i|w) dw \quad (4.48)$$

can be proven to have the following asymptotic expansion:

$$\begin{aligned} F_n = & nL_n(w_0) + \lambda \log n - (m-1) \log \log n \\ & - \log \left(\int dw D(w) \int_0^\infty dt t^{\lambda-1} \exp(-t + \sqrt{t} \xi_n(w)) \right) + o_p(1) , \end{aligned} \quad (4.49)$$

where:

$$L_n(w_0) = -\frac{1}{n} \sum_{i=1}^n \log p(X_i|w_0) . \quad (4.50)$$

Since λ and m depend on the data distribution q , one generally uses an estimator called *WBIC* which was introduced in [213]. This allows one to estimate F_n without any information about the true distribution [79] and is defined as:

$$\text{WBIC} := \mathbb{E}_w^\beta[nL_n(w)], \quad \beta = \frac{1}{\log n}$$

where $\mathbb{E}_w^\beta$ is the expectation over the posterior tempered by the inverse temperature β . Watanabe then proved that $\text{WBIC} = F_n + O_p(\sqrt{\log n})$, hence sharing the asymptotic expansion of the free energy while being achievable only from the sample alone.

Returning to the free energy formula, since leading order approximations drive macroscopic behavior, the lower order terms can be seen as corrections. The results summarized above allow one to approximate the free energy at large n as a sum of two terms:

$$F_n \approx nL_n(w_0) + \lambda \log n . \quad (4.51)$$

This is Watanabe's asymptotic approximant of the free energy.

4.4 Bayesian phase transitions

We define a phase α as a distinct local basin of attraction $W_\alpha \subset W$ for the KL divergence K in the parameter space W . Then W_α corresponds to a specific internal representation learned by the model. The investigation is done on $Z_{n,\alpha}$, as the posterior's probability mass is proportional to it and localized entirely on that specific phase [79].

From $F_{n,\alpha} = -\log Z_{n,\alpha}$ we obtain the usual description, with the added benefit that we can use the rules of statistical mechanics to assert that the Bayesian system under study will settle locally into the phase that minimizes $F_{n,\alpha}$.

The result obtained in the last section gives:

$$F_{n,\alpha} \approx nL_\alpha + \lambda_\alpha \log n , \quad (4.52)$$

where the phases are determined by the two competing terms:

- the energy term (nL_α) — a linear term in n which represent the accuracy of the model.
- the entropy penalty/complexity term ($\lambda_\alpha \log n$) — a term which grows logarithmically and penalizes phases with complex geometries.

The Bayesian phase transition is defined as the exact moment that the free energy shifts from one internal representation to another as the number of data n grows. The phase transition happens exactly at the dataset threshold n^* when the free energies between two phases are equal:

$$F_{n,A} = F_{n,B} \tag{4.53}$$

Now that we understand the methodology of how phases and phase transitions are described in Singular Learning Theory, we apply the framework to a simple neural network presented in the next chapter.

Toy models of superposition

This section contains our contribution to the SLT analysis of toy models of superposition [99]. After briefly explaining the relation of such models to superposition in ANNs (which justifies their name), we introduce them following Elhage’s paper and discuss some of the evidence for phase transitions in such models gathered from numerical experiments. We then explain their SLT analysis for the case $r = 2$ given in [100]. Finally, we analyze in detail the case $r = 1$, which was not analyzed before in the literature. Our results for that case constitute the original contribution of this thesis.

5.1 A primer on superposition in ANNs

The phenomenon of *superposition* in ANNs describes situations where an artificial neural network represents more features than it has available hidden dimensions by encoding those features into dimensionally compressed and linearly-dependent configurations of vectors. In large-scale neural networks, artificial neurons are polysemantic, meaning that single neuron might fire for a mixture of unrelated concepts (qualitatively distinct classes of data defined by collections of features).

Superposition explains why this happens: rather than assigning features to orthogonal coordinate axes in its representation space (which correspond to individual neurons), the ANN treats certain directions in this space as corresponding to features of the data. If a network tries to represent n features using only m hidden dimensions (where $m < n$), then it encodes those features as an overcomplete system of vectors in $\mathbb{R}^m$. Such compression tends to occur for feature-sparse data, which means that features of the data sample are weakly correlated in the empirical distribution. In that case, the internal representation of the data can involve a set of vectors which are not linearly independent without causing catastrophic interference. For such data, the network incurs small correlations between feature-representing vectors (which reproduce the weak correlations between the features of the dataset) but efficiently encodes a larger set of features than its hidden dimension would naively allow. We refer the reader to [99, 223, 224, 225, 226, 226, 227] for more information about superposition.

5.2 Introduction of TMS

The models we shall consider in detail are the *Toy Models of Superposition* (TMS) introduced in [99].

The main idea of a TMS is to analyze a small ReLU network where we can make sense of what happens inside. A type of artificial neural network that satisfies these properties and is the object of our study is the *simple autoencoder* — a feedforward network with a single hidden layer for which the widths of the input and output layer are equal to each other and larger than that of the hidden layer (which in this case is also called the “latent layer”). Such ANNs are used for encoding data features in a dimensionally-reduced (“compressed”) manner, being trained to reproduce the identity map as composition of their encoding and decoding maps¹ (see [228, 229, 230, 231] for more information about various types of autoencoders and their uses). We are interested in what happens inside the hidden layer, what representations the network creates and how they change during the learning process.

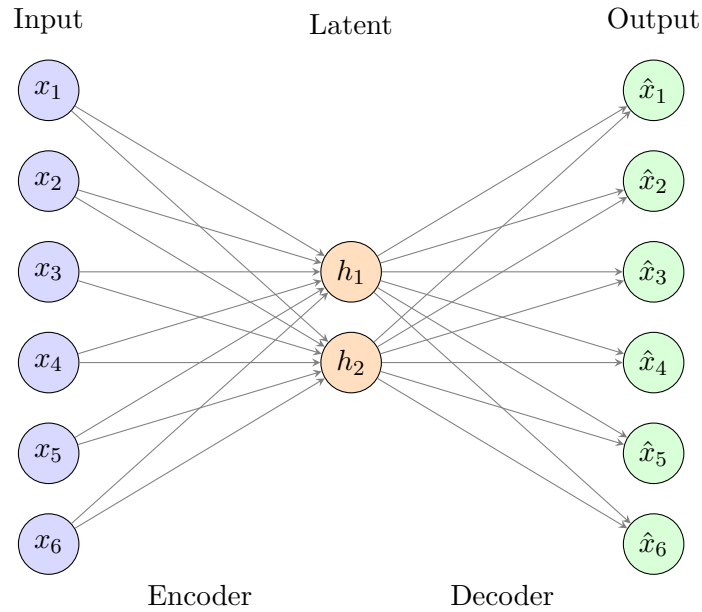


Figure 5.1: A 6-2-6 autoencoder with a two-dimensional latent bottleneck.

Let c be the common dimension of the input and output layers and r be the dimension of the hidden layer, where $r < c$. For the example shown in figure 5.1 we have $c = 6$ and $r = 2$. We assume for simplicity that the neurons of the input and output layer have zero biases and trivial activation functions (equal to the identity map), while the hidden layer neurons have ReLU activation functions and nontrivial biases b_i ($i = 1, \dots, c$), which form the bias vector $b \in \mathbb{R}^c$. We also assume that weights are mirrored across the network. With these assumptions, the input, middle and output layer implement the maps:

$$\mathbb{R}^c \ni x \rightarrow h = Wx \in \mathbb{R}^r, \quad \mathbb{R}^r \ni h \rightarrow \hat{x} = \text{ReLU}(W^T h + b) \in \mathbb{R}^c, \quad \mathbb{R}^c \ni \hat{x} \rightarrow \hat{x} \in \mathbb{R}^c. \quad (5.1)$$

¹The input layer produces the encoding map from data vectors to vectors of the hidden layer space (often called *latent vectors*). The output layer decodes the latent vectors into output vectors.

For the second map above, the notation means that the ReLU activation function acts on each of the components of the vector $W^T h + b \in \mathbb{R}^c$. The weight matrix $W \in \text{Mat}_{r,c}(\mathbb{R})$ is called the *encoding matrix*, while its transpose $W^T \in \text{Mat}_{r,c}(\mathbb{R})$ is the *decoding matrix*. With these assumptions, the full autoencoder implements the composition of the three maps above:

$$\mathbb{R}^c \ni x \rightarrow \hat{x} = \text{ReLU}(W^T W x + b) \in \mathbb{R}^c . \quad (5.2)$$

In the generic case when W has maximal rank (equal to r), the activation space of hidden layer neurons is spanned by its columns $W_i \in \mathbb{R}^r$, which define directions which represent features of sparse datasets [99]. Accordingly, we shall refer to these columns as *latent feature vectors*.

As an example, we reproduce a numerical experiment from [99] which illustrates how features appear in superposition as sparsity S is increased.

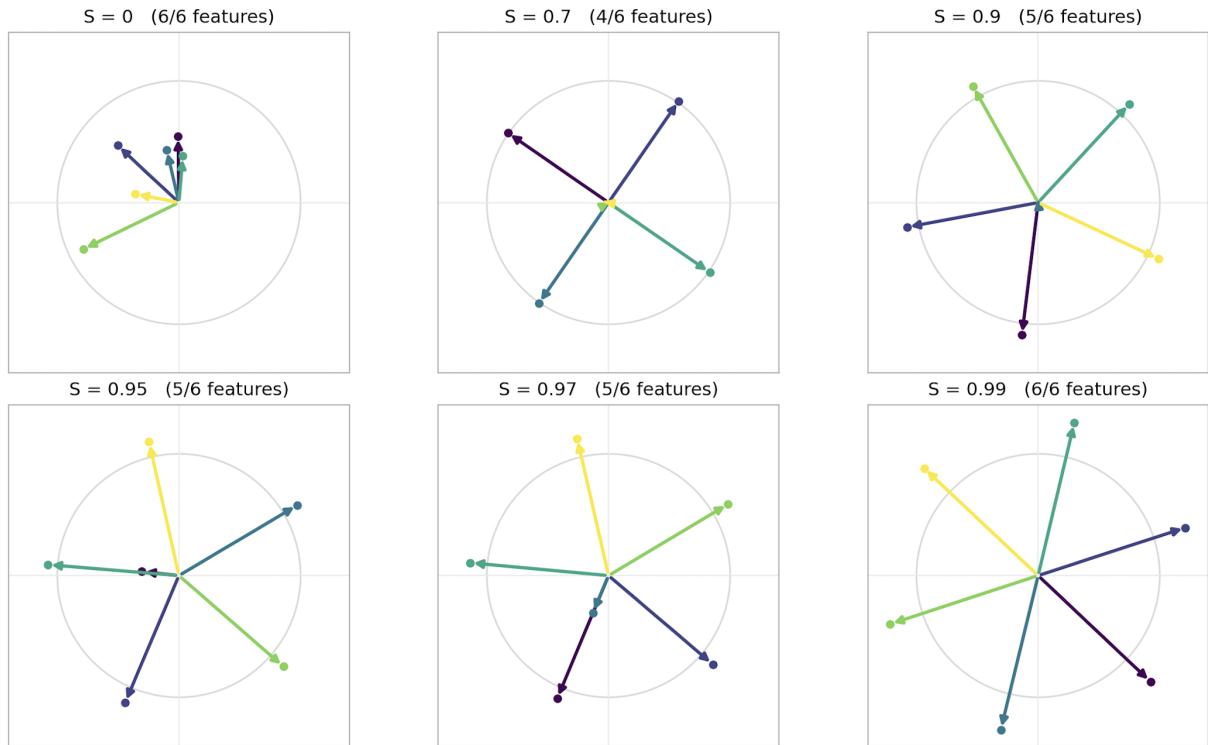


Figure 5.2: *Emergence of superposition in a toy ReLU autoencoder (after [99]) showing how the inner representation of features through latent vectors changes as sparsity is increased.*

When the dataset is non-sparse, the figure illustrates how the autoencoder tends to arrange all latent feature vectors along two orthogonal directions, being unable to distinguish more than two features. As data sparsity increases, the penalty of correlations between feature representations drops and superposition starts to take place. For high sparsity, the preferred geometric arrangements of latent feature vectors arise from regular polygons. We discuss this in the next chapter where we explain why regular polygon configurations have physical meaning and drive the learning process of the network.

For $r > 2$, it was found empirically that increasing sparsity leads to arrangements of latent feature vectors which correspond to different types of regular polyhedra. The network

seems to prefer such emergent structures due to their enhanced rotational symmetry (encoded by their stabilizer inside the rotation group). Moreover, the training loss was found empirically to decrease in abrupt steps as different such polyhedra are formed in the representation of the neural network, which suggested that the network undergoes *phase transitions*. This is the phenomenon that we want to study rigorously from an SLT perspective, where phase Bayesian phase transitions in neural networks are mathematically established.

5.3 SLT analysis for the case $r = 2$

This section follows reference [100], which analyzed the $r = 2$ class of TMS networks through the lens of Singular Learning Theory. They first derived a closed formula for the theoretical loss, then proved that the k -gon (regular polygon) configurations of latent feature vectors correspond to critical points of the free energy and that the local learning coefficient introduced in [217] determines the Bayesian phase transitions. Finally, they showed empirically that the Stochastic Gradient Descent (SGD) learning trajectory of such models is driven to such configurations, thus formulating the so-called “Bayesian antecedent hypothesis” (BAH), which states that Bayesian phase transitions as described in SLT cause superposition and more generally structure formation in ANNs.

Let $X := [0, 1]^c$ and $\mathcal{W} \subseteq \text{Mat}_{r,c}(\mathbb{R}) \times \mathbb{R}^c$ be the parameter space, whose elements are the pairs $w = (W, b)$ consisting of a weight matrix $W \in \text{Mat}_{r,c}(\mathbb{R})$ and a bias vector $b \in \mathbb{R}^c$. We assume that $\mathcal{W}$ is compact and satisfies the conditions of Chapter 4 together with the relative finite variance condition.

A simple autoencoder as considered in the previous section is completely described by the map:

$$\begin{aligned} f : X \times \mathcal{W} &\longrightarrow \mathbb{R}^c, \\ f(x, w) &= \text{ReLU}(W^T W x + b) , \end{aligned} \tag{5.3}$$

where $w = (W, b) \in \mathcal{W} \times \mathbb{R}^c$ and x is constrained to belong to the bounded set $X \subset \mathbb{R}^c$ to ensure compactness of the set of allowed data vectors. With these notations, the compact set X is the topological space of data vectors, while $\mathcal{W}$ is the space of parameters of the model.

The data generating distribution is taken to be [100]:

$$q(x) = \sum_{i=1}^c \frac{1}{c} \delta_{x \in C_i} = \sum_{i=1}^c \frac{1}{c} \delta(x_i) , \tag{5.4}$$

where C_i denotes the i -th coordinate axis intersected with X . This corresponds to the average of the Dirac measures of the segments $C_1, \dots, C_c$ in the c -dimensional cube X . We use the parameterized Bayesian model defined by the likelihood function [100]:

$$p(x|w) \propto \exp \left(-\frac{1}{2} \|x - f(x, w)\|^2 \right) . \tag{5.5}$$

This penalizes exponentially deviations of the parameterized function $x \rightarrow f(x, w)$ from the identity function of X using the penalty function $\Omega : X \times X \rightarrow \mathbb{R}_{\geq 0}$ given by:

$$\Omega(x, y) = \|x - y\|^2 . \tag{5.6}$$

The likelihood function (5.5) is chosen because autoencoders are trained to reproduce the input data.

We take the prior $\varphi(w)$ to be a density on $\mathcal{W}$ satisfying the conditions of Chapter 4 together with the relative finite variance condition. Its precise form does not matter for the leading-order analysis, since the prior affects only the $O(1)$ term of the free energy provided $\varphi(w_0) > 0$. In [100], the standard normal distribution $\mathcal{N}(0, 1)$ is used as the prior for sampling the posterior.

The triple $(p(x|w), q(x), \varphi(w))$ is assumed, with our choice of penalty function and data generating distribution, which gives the population loss function:

$$L(w) = \int_{\mathcal{X}} q(x) \|x - f(x, w)\|^2 dx . \quad (5.7)$$

Since the ReLU activation function is not smooth, we need to partition the parameter space $\mathcal{W}$ into “safe subsets” where various relevant functions are differentiable. For that purpose, we consider the following subsets [100]:

1. $P_{i,j} = \{(W, b) \in \text{Mat}_{r,c}(\mathbb{R}) \times \mathbb{R}^c \mid W_i \cdot W_j > 0 \text{ and } -W_i \cdot W_j \leq b_i \leq 0\}$;
2. $P_i = \{(W, b) \in \text{Mat}_{r,c}(\mathbb{R}) \times \mathbb{R}^c \mid \|W_i\|^2 > 0 \text{ and } -\|W_i\|^2 \leq b_i \leq 0\}$;
3. $Q_{i,j} = \{(W, b) \in \text{Mat}_{r,c}(\mathbb{R}) \times \mathbb{R}^c \mid -W_i \cdot W_j > b_i > 0\}$

where $W \in \text{Mat}_{r,c}(\mathbb{R})$ and $W_1, \dots, W_c$ are the columns of W . We use the notation $\delta(M)$ for the characteristic function of a subset M of $\mathcal{W}$ and a similar notations for subsets of $\mathbb{R}$. [100]

Lemma 3.1. from [100] gives the population loss within these three zones as $L(w) = \frac{1}{3c} H(w)$, where:

$$H(W, b) = \sum_{i=1}^c \delta(b_i \leq 0) H_i^-(W, b) + \delta(b_i > 0) H_i^+(W, b) \quad (5.8)$$

and

$$\begin{aligned} H_i^-(W, b) &= \sum_{j \neq i} \delta(P_{i,j}) \left[\frac{1}{W_i \cdot W_j} (W_i \cdot W_j + b_i)^3 \right] \\ &\quad + \delta(P_i) \left[\frac{b_i^3}{\|W_i\|^4} + \frac{b_i^3}{\|W_i\|^2} \right] + (1 - \delta(P_i)) + \delta(P_i) N_i \\ H_i^+(W, b) &= \sum_{j \neq i} \delta(Q_{i,j}) \left[-\frac{1}{W_i \cdot W_j} b_i^3 \right] \\ &\quad + \sum_{j \neq i} (1 - \delta(Q_{i,j})) \left[(W_i \cdot W_j)^2 + 3(W_i \cdot W_j) b_i + 3b_i^2 \right] + N_i , \end{aligned} \quad (5.9)$$

where

$$N_i = (1 - \|W_i\|^2)^2 - 3(1 - \|W_i\|^2) b_i + 3b_i^2. \quad (5.10)$$

From this analysis the authors of [100] calculated the local learning coefficient (llc) introduced in [217], finding that k-gons correspond at best to minimally singular configurations,

so a naive dimension count gives the correct llc. No desingularization is needed in this case. By choosing an $r = 2, c = 6$, they analyzed the Bayesian phase transitions in the corresponding TMS.

A dataset $\mathcal{D}_n = \{x_1, x_2, \dots, x_n\}$ has empirical negative log likelihood function $L_n(w) = -\frac{1}{n} \sum_{i=1}^n \log p(x_i | w)$ [100]. Using the free energy formula explained in chapter 4, let $\{\mathcal{W}_\alpha\}_\alpha$ be a finite cover of $\mathcal{W}$ by compact semi-analytic subsets of $\mathcal{W}$ with nonempty interior. Assume that each of the sets $\mathcal{W}_\alpha$ contains a point w_α^* which minimizes L and that the restriction of our Bayesian model to each $\mathcal{W}_\alpha$ satisfies the relative finite variance conditions. In this case, [100] showed that the local free energy on each set of such a cover of $\mathcal{W}$ has the following asymptotic behavior for large n :

$$F_n(\mathcal{W}_\alpha) = nL_n(w_\alpha^*) + \lambda_\alpha \log n - (m_\alpha - 1) \log \log n + O_p(1) , \quad (5.11)$$

where the subleading terms can be approximated by real constants c_α . The global free energy is well-approximated for large n as:

$$F_n \approx \min_\alpha [nL_n(w_\alpha^*) + \lambda_\alpha \log n + c_\alpha] . \quad (5.12)$$

The asymptotic forms above imply the existence of local phase transitions as n grows towards infinity (which corresponds to the zero temperature limit). Such local phase transitions become global if there exists some value n_{cr} of n for which the contribution of $F_n(\mathcal{W}_\alpha)$ dominates all others for $n \geq n_{cr}$.

With the theory ready, they analyzed the behavior of stochastic gradient descent trajectories for large n and found that almost all transitions which are observed numerically have a Bayesian antecedent, with a single exception which was numerically inconclusive.

In the next section, we study in detail the case $r = 1$, which was never analyzed before in the literature.

5.4 Study of the case $r = 1$

This section and the next present the original contribution of this thesis. We first perform a mathematically rigorous study of toy models of superposition in the case $r = 1$. Finding the critical points of the free energy case by case, we derive a full theoretical classification, describe the types of such critical points and calculate the corresponding llc where possible. We also discuss the stability of the configuration of latent feature vectors for each type of critical point and check if a phase transition can occur. Throughout our analysis, we make the same data sparsity and technical assumptions as for the $r = 2$ case discussed above (see [100] and the previous section).

We start from the dimension-agnostic formula for the TMS potential, which we shall specialize to the case $r = 1$:

$$H(W, b) = \sum_{i=1}^c \delta(b_i \leq 0) H_i^-(W, b) + \delta(b_i > 0) H_i^+(W, b) \quad (5.13)$$

where:

$$H_i^-(W, b) = \sum_{j \neq i} \delta(P_{i,j}) \left[\frac{1}{W_i \cdot W_j} (W_i \cdot W_j + b_i)^3 \right] + \delta(P_i) \left[\frac{b_i^3}{\|W_i\|^4} + \frac{b_i^3}{\|W_i\|^2} \right] + (1 - \delta(P_i)) + \delta(P_i) N_i \quad (5.14)$$

$$H_i^+(W, b) = \sum_{j \neq i} \delta(Q_{i,j}) \left[-\frac{1}{W_i \cdot W_j} b_i^3 \right] + \sum_{j \neq i} (1 - \delta(Q_{i,j})) \left[(W_i \cdot W_j)^2 + 3(W_i \cdot W_j) b_i + 3b_i^2 \right] + N_i \quad (5.15)$$

and N_i is given by:

$$N_i = (1 - \|W_i\|^2)^2 - 3(1 - \|W_i\|^2) b_i + 3b_i^2. \quad (5.16)$$

For the case $r = 1$, we have $W \in \text{Mat}_{1,c}(\mathbb{R})$ and $b \in \mathbb{R}^c$. The total parameter dimension is $rc + c = 2c$, which is the total number of learnable parameters in our case.

When $r = 1$, the space of latent vectors is one-dimensional and hence all latent feature vectors lie on the real axis pointing either on the left or to the right. For each i we can write $W_i = s_i l_i$ with $l_i \geq 0$ and $s_i \in \{+1, -1\}$. The sign factor s_i is defined only when $l_i > 0$, as the sign is undefined when $l_i = 0$.

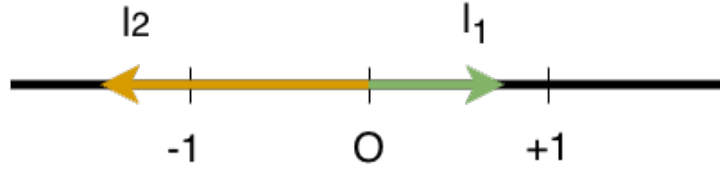


Figure 5.3: $r = 1$ example diagram of two represented features

The dot product of vectors reduces to the ordinary multiplication of real numbers:

$$W_i \cdot W_j = W_i W_j = s_i s_j l_i l_j$$

while the square norm of latent feature vectors reduces to:

$$\|W_i\|^2 = l_i^2.$$

Let us classify the symmetries specific to the case $r = 1$, which allow us to further simplify and analyze the TMS potential:

- **orthogonal symmetry:** The orthogonal group in one dimensions $O(1) = \{+1, -1\}$ acts by global sign flip $W \rightarrow -W$ and is a discrete symmetry group ($O(1)$ can be viewed as zero-dimensional Lie group).
- **permutation symmetry:** We have $H(W_\pi, b_\pi) = H(W, b)$, for any $\pi \in S_c$, where S_c denotes the group of permutations of c objects.

Due to these symmetries, for c features we must have $c! \times 2$ mathematically identical copies of every state. We are interested in transformations of the network's weights that change their values but preserve the relative geometry (the dot products). This ensures that such transformations preserve the form of the potential.

We next specify the chamber structure in the one-dimensional case which is analogous to the two-dimensional wedge defined in [100].

By definition, we have $W_i \cdot W_j > 0$ if and only if $s_i \cdot s_j = +1$. This is the condition that W_i and W_j share sign. Hence in our case the chambers are determined by the sign partition:

$$\begin{cases} S_+ = \{i : s_i = +1\} \\ S_- = \{i : s_i = -1\} \end{cases}, \quad (5.17)$$

where S_+ contains all latent feature vectors which point to the right and S_- all those which point to the left. As long as latent vectors stay within a specific chamber, the rules of what features interfere with each other are unchanged.

Any two feature vectors lying in the same sign partition will have a positive dot product, so they will interfere with one another and drive up the loss function. This holds up to a global sign flip due to the global $O(1)$ symmetry.

We use graphs in order to keep track of which latent feature vectors are correlated, where every feature from 1 to c is represented by a node and if two features are correlated then we draw an edge between them. So the chamber graph C is the bipartite-complement graph where we draw lines connecting every node within the same group, but draw zero lines between the groups. This is a helpful way of thinking about this graphical representation, which we can write concretely as the rule:

$$\{i, j\} \in C \iff s_i \cdot s_j = +1. \quad (5.18)$$

Therefore, for the one dimensional case, instead of making up a wedge table we ask which columns share the same sign.

As a simple example, if we have $S_+ = \{1, 2, 3\}$ and $S_- = \{4, 5\}$, then:

$$C = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{4, 5\}\}$$

and as long as we don't cross the origin of the real axis, the set we just built is "frozen" and we are trapped in the same chamber of the parameter space.

We now study case by case the TMS potential in different safe zones, in order to identify and classify the critical points.

5.4.1 The case $c = k$ with $b_i \leq 0$

We restrict our attention to the case when $b_i \leq 0$ for all i , which is analogous to that considered in [100, Section H.1].

Lemma 1. *Let C correspond to a sign partition $\{S_+, S_-\}$ and let $(W, b) \in S(C)$ (the well-behaved region defined by this configuration) with $b_i \leq 0$ for all i . Then:*

$$H(W, b) = \sum_{i=1}^c \left[\sum_{j \in S_{\sigma(i)}, j \neq i} \frac{(l_i l_j + b_i)^3}{l_i l_j} + \frac{b_i^3}{l_i^4} + \frac{b_i^3}{l_i^2} + (1 - l_i^2)^2 - 3(1 - l_i^2)b_i + 3b_i^2 \right], \quad (5.19)$$

where $\sigma(i) \in \{+, -\}$ is the sign of the column i .

Proof. Restricting to the $b_i \leq 0$ for all i , the positive bias term from the potential vanishes entirely and we are left with:

$$H(W, b) = \sum_{i=1}^c H_i^-(W, b). \quad (5.20)$$

As explained in the beginning of the section, we parameterize $W_i = s_i l_i$ in the one dimensional case and hence:

- dot product: $W_i \cdot W_j = s_i s_j l_i l_j$
- the squared norm: $\|W_i\|^2 = l_i^2$.

We are given a chamber graph C defined by the sign partition $\{S_+, S_-\}$. If the column i is in the partition given by $S_{\sigma(i)}$, then:

- For any $j \in S_{\sigma(i)}$, their signs match $s_i \cdot s_j = +1$ so $W_i \cdot W_j = l_i \cdot l_j > 0$
- For any $j \notin S_{\sigma(i)}$, their signs differ $s_i \cdot s_j = -1$ so $W_i \cdot W_j = -l_i \cdot l_j < 0$.

The indicator function $\delta(P_{i,j})$ is equal to 1 if $W_i \cdot W_j > 0$ and $-W_i \cdot W_j \leq b_i \leq 0$. Because of our sign partition:

- If $j \notin S_{\sigma(i)}$, then $W_i \cdot W_j < 0$ so the indicator $\delta(P_{i,j}) = 0$ and the term vanishes entirely.
- If $j \in S_{\sigma(i)}$, then $W_i \cdot W_j = l_i \cdot l_j > 0$. Because we assume $(W, b) \in S(C)$, we are well within the support of the chamber where $-l_i l_j \leq b_i$ holds, so $\delta(P_{i,j}) = 1$.

Therefore, the sum over all $j \neq i$ collapses to a sum over only the features that are in the same sign partition:

$$\sum_{j \neq i} \delta(P_{i,j})[\dots] \Rightarrow \sum_{j \in S_{\sigma(i)}, j \neq i} \frac{(l_i l_j + b_i)^3}{l_i l_j}.$$

The indicator $\delta(P_i)$ equals 1 if $\|W_i\|^2 > 0$ and $-\|W_i\|^2 \leq b_i \leq 0$. Since we assume that (W, b) is a valid configuration inside the chamber — meaning that all the lengths $l_i > 0$ and we haven't violated the bias boundaries —, we have $\delta(P_i) = 1$. So $1 - \delta(P_i) = 0$, $\|W_i\|^2 = l_i^2$ and $\delta(P_i)N_i \rightarrow N_i$, which give:

$$H(W, b) = \sum_{i=1}^c \left[\sum_{j \in S_{\sigma(i)}, j \neq i} \frac{(l_i l_j + b_i)^3}{l_i l_j} + \frac{b_i^3}{l_i^4} + \frac{b_i^3}{l_i^2} + (1 - l_i^2)^2 - 3(1 - l_i^2)b_i + 3b_i^2 \right]$$

thus completing the proof. $\square$

We also use the notations $|S_+| = k_+$, $|S_-| = k_-$, with k the number of features actually represented by the network and k_+, k_- are the number of vectors in each sign partition. In this particular case, we have $k_+ + k_- = c$.

By the above result, the sum splits cleanly: the positive sign columns interact with each other but not with the negative sign columns. So $H = H_+ + H_-$ where $H_\pm$ depends only on the columns of sign $\pm$.

Using the symmetry argument, we stress that once we lock in a specific configurations we can no longer shuffle every vector all at once, because if we swap a positive feature with a negative one, we change the graph. However, we can still shuffle the positive features among themselves and the negative feature among themselves. The leftover shuffling we are allowed to perform is described by the group $S_{k_+} \times S_{k_-}$, which gives the residual permutation symmetry.

To move further along in finding the critical points of this problem we need to simplify it using the natural Ansatz that if a system has perfect symmetry, then its optimal stable configuration (the critical points) will almost always adopt the exact same symmetry.

Following this Ansatz, we can make all the positive sign columns identical and all negative-sign columns identical:

$$\begin{aligned} l_i &= l_+ & \text{for } i \in S_+, & & b_i &= y_+ & \text{for } i \in S_+, \\ l_i &= l_- & \text{for } i \in S_-, & & b_i &= y_- & \text{for } i \in S_-. \end{aligned} \quad (5.21)$$

Under the symmetric Ansatz, we can write H as a sum of two contributions:

$$H = k_+ h(l_+, y_+; k_+) + k_- h(l_-, y_-; k_-) , \quad (5.22)$$

where we can write h as a general function:

$$h(l, y; k) = \frac{(k-1)(l^2 + y)^3}{l^2} + \frac{y^3}{l^4} + \frac{y^3}{l^2} + (1-l^2)^2 - 3(1-l^2)y + 3y^2 \quad (5.23)$$

which can be used for each of the case $\{+, -\}$ and depends on the $k-1$ neighbors of the sign partition. To find the critical points, we need to check $\nabla H = 0$, where each block $\pm$ optimizes itself in complete isolation. Hence to find the optimal values for the positive and negative lengths and biases, we write:

$$\begin{aligned} \frac{\partial H}{\partial l_+} &= 0, & \frac{\partial H}{\partial y_+} &= 0, \\ \frac{\partial H}{\partial l_-} &= 0, & \frac{\partial H}{\partial y_-} &= 0 . \end{aligned} \quad (5.24)$$

This also reads:

$$\begin{aligned} \frac{\partial h(l_+, y_+; k_+)}{\partial l_+} &= 0, & \frac{\partial h(l_+, y_+; k_+)}{\partial y_+} &= 0, \\ \frac{\partial h(l_-, y_-; k_-)}{\partial l_-} &= 0, & \frac{\partial h(l_-, y_-; k_-)}{\partial y_-} &= 0 . \end{aligned} \quad (5.25)$$

Using h as a general template, we take the derivatives with respect to l and y and require them to vanish, which gives:

$$y \left(1 + \frac{y}{l^2} \right)^2 = 0 . \quad (5.26)$$

Here we impose the following conditions, which ensure that we obtain physically valid answers:

- $l > 0$
- $-l^2 \leq y \leq 0$.

Solving the equation gives the following valid critical points:

- **An interior critical point** — [1] h^* for $(y = 0, l = \frac{1}{\sqrt{k}})$
- **A chamber boundary critical point** — [2] h^* for $(y = -l^2, l)$.

We find the value of [1] to be $h^* = 1 - \frac{1}{k} = \frac{k-1}{k}$, while for [2] we obtain $h^* = 1$.

To find all possible combinations, we mix these two and calculate the potential and loss. Inserting them in the general formula gives:

- [1][1] $\Rightarrow H^* = k_+ + k_- - 2$
- [1][2] $\Rightarrow H^* = k_+ + k_- - 1$
- [2][1] $\Rightarrow H^* = k_+ + k_- - 1$
- [2][2] $\Rightarrow H^* = k_+ + k_-$.

We next calculate the loss using the general formula:

$$L = \frac{H^*}{3c} , \quad (5.27)$$

which gives:

- [1][1] $\Rightarrow L = c - 2$
- [1][2] $\Rightarrow L = c - 1$
- [2][1] $\Rightarrow L = c - 1$
- [2][2] $\Rightarrow L = c$.

We can now compute L for various values of $c = k$ and index the different values in a table:

c	configuration	(k_+, k_-)	H^*	L
1	all-same-sign interior	$(1, 0)$	0	0
1	all-same-sign boundary	$(1, 0)$	1	$1/3$
2	all-same-sign interior	$(2, 0)$	1	$1/6$
2	all-same-sign boundary	$(2, 0)$	2	$1/3$
2	$(1, 1)$ both interior	$(1, 1)$	0	0
2	$(1, 1)$ mixed (one int., one bdy.)	$(1, 1)$	1	$1/6$
2	$(1, 1)$ both boundary	$(1, 1)$	2	$1/3$
3	all-same-sign interior	$(3, 0)$	2	$2/9$
3	all-same-sign boundary	$(3, 0)$	3	$1/3$
3	$(2, 1)$ both interior	$(2, 1)$	1	$1/9$
3	$(2, 1)$ mixed	$(2, 1)$	2	$2/9$
3	$(2, 1)$ both boundary	$(2, 1)$	3	$1/3$
4	$(1, 3)$ or $(3, 1)$ both interior		2	$1/6$
4	$(2, 2)$ both interior		2	$1/6$
4	all-same-sign interior	$(4, 0)$	3	$1/4$

Table 5.1: *The loss for different configurations with $c = k$*

For the all-boundary critical points we find a loss value of $\frac{1}{3}$, which represents the situation where the latent bias vectors become very large, effectively "killing the vectors". This is the value of a completely dead, non-functioning autoencoder.

We calculated the Hessian numerically and found only two non-degenerate local minima, so only two configurations are valid for calculating the llc (with a naive dimension count). The rest of them are saddles, not any of them minimally singular, due to the negative eigenvalues that appeared when checking.

The local learning coefficient is calculated using:

$$\lambda = \frac{2c - \dim(\text{level set})}{2}, \quad (5.28)$$

where $\dim(\text{level set})$ is given by the type of critical point. The llc gives a real learning coefficient only at minimum or minimally-singular critical points (positive-definite Hessian modulo the level set kernel). The only two configurations which admit an llc are:

c	configuration	L	λ
1	(1,0) interior	0	1
2	(1,1) both interior	0	2

Table 5.2: *Valid llc 's in the two configurations for the case $c = k$*

Some **conclusions** for the case $r = 1$ with $c = k$:

- For $c = 1$ — trivially, one column maps one feature in the one dimensional hidden space and ReLU reproduces it. We have $L = 0$, $\lambda = 1$, which is a genuine non-degenerate global minimum.
- For $c = 2$ — there are three distinct H-values for the symmetric Ansatz critical points: $H \in \{0, 1, 2\}$, but only the (1,1) -split-both-interior is a local minimum, with $L = 0$ and $\lambda = 2$. All others are saddles. This configuration has just enough capacity (two sign directions, two features) to achieve perfect auto-encoding, with a positive-definite Hessian at the minimum.
- For $c \geq 3$ all symmetric Ansatz critical points are saddles.

Notice that in this case in order to have true superposition we would have to cram c features into r dimensions when $c > 2r$, which does not give us valid critical points. Hence when we study this case, we cannot state or do analysis for $c \geq 3$, which would give anything meaningful. We investigate further the case $r = 1$ case is done in the next subsections, where we allow vestigial columns and see what optimal configuration the network chooses.

5.4.2 The case of negative bias for $k < c$

For this case, the columns split into three categories:

- active positive - their value W_i is a positive number
- active negative - their value W_i is a positive number
- vestigial - their value is exactly 0

We call these three groups, respectively, S_+ , S_- and V , with sizes:

$$|S_+| = k_+ \tag{5.29}$$

$$|S_-| = k_- \tag{5.30}$$

$$|V| = c - k \tag{5.31}$$

and the condition $k = k_+ + k_-$. Thus, the analysis for the case $k < c$ — at least one vestigial column exists. If this is not the case and all columns are represented, we return to the case of $k = c$, the case we already studied.

The dot products reduces as follows, were we group based on the sign and if they are represented or not:

$$W_i \cdot W_j = \begin{cases} +l_+l_+ = l_+^2 & i, j \in S_+ \\ +l_-l_- = l_-^2 & i, j \in S_- \\ -l_+l_- & i \in S_+, j \in S_- \\ 0 & i \in V \text{ or } j \in V \end{cases} \quad (5.32)$$

For the:

- active positive columns ($i \in S_+$) we find $h(l_+, y_+; k_+)$
- active negative columns ($i \in S_-$) we find $h(l_-, y_-; k_-)$
- vestigial — $i \in V$ so $W_i = 0$, hence every $W_i \cdot W_j = 0$ and because it isn't strictly positive we have no chamber inclusion and $\delta(P_{i,j}) = 0$. Also the self-bias term requires $\|W_i\|^2 > 0$ which fails and the entire formula reduces to $H_i = (1 - \delta(P_i)) = 1$

Each vestigial column contributes exactly the constant 1 to H regardless of what b_i is, provided that $b_i \leq 0$, which means that b_i is a flat direction of H for the vestigial column. Nothing else changes from the $c = k$ analysis up until here and our potential for this case can be written as:

$$H = k_+ h(l_+, y_+; k_+) + k_- h(l_-, y_-; k_-) + (c - k) \quad (5.33)$$

Due to the form of H, we can asses the following in the exact same style, following [100]:

Let $H_{active}^{(k)}(l_+, y_+, l_-, y_-) = k_+ h(l_+, y_+; k_+) + k_- h(l_-, y_-; k_-)$ denote the potential of the active $c = k$ case. Let $\Pi_{c,k}$ be the projection of the full (W, b) space onto the active represented features. Then in an open neighborhood of the configuration with $W_i = 0$ for $i \in V$, $W_i = l_\pm$ for $i \in S_\pm$, and any negative b_i for $i \in V$:

$$H(W, b) = H_{active}^{(k)} \circ \Pi_{c,k}(W, b) + (c - k) . \quad (5.34)$$

To find the critical points we take the derivatives as before and additionally for the vestigial columns:

$$\frac{\partial H}{\partial b_i} = 0 \quad (5.35)$$

for $i \in V$, which is automatically satisfied and represents a flat direction, and

$$\frac{\partial H}{\partial W_i} = 0 \quad (5.36)$$

for $i \in V$ and at $W_i = 0$, all terms vanish to leading order, so the gradient is zero in this direction. This means that the critical points are completely determined by the active sub-group choices, and the vestigial columns are passive.

We now focus on the Hessian analysis in order to find which critical points check all conditions, so that we can calculate their complexity. The Hessian of H at any of these critical points is block-diagonal with three blocks: the S_+ block, the S_- block and the vestigial V block. This is because the three groups don't couple in H .

- The S_+ block has the form $2k_+ \times 2k_+$ and further decomposes for the interior critical point into a symmetric direction with two strictly positive eigenvalues and an antisymmetric direction (multiplicity $k_+ - 1$) with one positive and one negative eigenvalue. This gives $k_+ - 1$ negative eigenvalues. For the boundary critical point, the symmetric direction has eigenvalues 0 (tangent along the boundary curve) and a strict negative value (transverse). The antisymmetric directions here only add more negative values.
- The S_- block is similar to the S_+ block because of the symmetry.
- The vestigial block ($2(c - k) \times 2(c - k)$) — for each vestigial column i :
 - The b_i direction is flat: $\frac{\partial^2 H}{\partial b_i^2} = 0$. The tangent direction.
 - The W_i direction: at $W_i = 0$, perturbing by ϵ activates configurations with one of the active sub-groups (positive sub-groups if $\epsilon > 0$, negative sub-group if $\epsilon < 0$).
 - If both $k_+ \geq 1$ and $k_- \geq 1$ then $\Delta H = \epsilon^2$ from either side. The function is C^2 at $W_i = 0$ and $\frac{\partial^2 H}{\partial W_i^2} = 2 > 0$ with positive eigenvalues.
 - If only one sub-group is active (say $k_+ \geq 1, k_- = 0$) then $\Delta H = \epsilon^2$ for $\epsilon > 0$, but $\Delta H = 0$ for $\epsilon < 0$. The function is C^1 but not C^2 at $W_i = 0$. This analysis tells us that the Hessian is directional and this is a chamber-wall non-analyticity.

We now analyze the tangent space of critical points (the "free" dimensions in the network at any critical point) and the minimally singular condition. The tangent space at a given configuration is the union between:

1. The tangent space contribution from each active sub-group's critical point type:
 - 0-dim interior critical point
 - 1-dim boundary critical point along $y = -l^2$
2. The vestigial bias freedoms, 1-dim per vestigial columns, totaling $c - k$ dim.

Thus:

$$\dim(\text{tangent}) = (c - k) + \#\{\text{sub-groups at boundary}\} . \quad (5.37)$$

In order for each configuration to be minimally singular, the Hessian needs to be positive-definite when restricted to the normal complement. Therefore the configuration is minimally singular when:

1. Every active sub-group has size one (no antisymmetric saddle).
2. Every active sub-group is at the interior critical point (no boundary saddle).
3. Both sub-groups are non-empty (so the vestigial W_i direction is C^2).

For $k_+ + k_- = k \leq 2$, with both ≥ 1 , condition (3) forces $k_+ = k_- = 1$ so that condition (1) is true and condition (2) gives the unique minimum configuration:

$$(1, 1) + (c - 2) \text{ standard vestigials} . \quad (5.38)$$

Thus for any $c \geq 2$, the configuration:

$$W_1 = +1, \quad W_2 = -1, \quad W_i = 0 \ (i = 3, \dots, c), \quad b_1, b_2 = 0, \quad b_i \in (-\infty, 0) (i \geq 3)$$

is the minimally-singular local minimum of H when $r = 1$. Its values are:

$$H^* = c - 2 \quad (5.39)$$

$$L^* = \frac{c - 2}{3c} . \quad (5.40)$$

Analyzing the Hessian at this point, we have $c + 2$ strictly positive eigenvalues, $c - 2$ zero eigenvalues — which match the $c - 2$ flat vestigial bias directions —, zero negative eigenvalues so the Hessian is positive-definite.

Because the configuration is minimally singular, the dimension count llc formula applies:

$$\lambda = \frac{1}{2}(\dim(\text{parameter space}) - \dim(\text{level-set tangent})) = \frac{1}{2}(2c - (c - 2)) = \frac{c + 2}{2}. \quad (5.41)$$

Specifically:

c	configuration	L_*	tangent dim	λ
2	(1, 1), no vestigial	0	0	2
3	(1, 1) + 1 vestigial	$1/9 \approx 0.111$	1	$5/2$
4	(1, 1) + 2 vestigials	$1/6 \approx 0.167$	2	3
5	(1, 1) + 3 vestigials	$1/5 = 0.200$	3	$7/2$
6	(1, 1) + 4 vestigials	$2/9 \approx 0.222$	4	4
7	(1, 1) + 5 vestigials	$5/21 \approx 0.238$	5	$9/2$
8	(1, 1) + 6 vestigials	$1/4 = 0.250$	6	5

Table 5.3: *Negative bias case, $k < c$, loss and llc values*

The configuration $(1, 0) + (c - 1)$ vestigial columns requires a directional analysis, because only one sub-group is active and we have the case of non-analyticity.

5.4.3 The case of positive bias on vestigial columns

We consider the configuration of $k_+ = k_- = 1$ active columns plus $c - 2$ vestigial columns as this is a mathematical safe spot. Unlike the previous case, we now allow for some of the vestigial columns to have a positive bias $b_i = 0$.

Let $0 \leq \sigma \leq c - 2$ be the number of " σ -positive" vestigial columns. The configuration is:

- $W_1 = 1, b_1 = 0$ (active positive)
- $W_2 = -1, b_2 = 0$ (active negative)
- For σ -vestigial columns: $W_i = 0, b_i = b^* > 0$ (to be determined)
- For the remaining $c - 2 - \sigma$ vestigial columns: $W_i = 0, b_i < 0$ arbitrary (standard vestigial)

For a vestigial column i with $W_i = 0$ and $b_i = b > 0$, we need to use the H_i^+ branch of Lemma 3.1. from the original paper:

$$H_i^+(W, b) = \sum_{j \neq i} \delta(Q_{i,j}) \left[-\frac{1}{W_i \cdot W_j} b_i^3 \right] + \sum_{j \neq i} (1 - \delta(Q_{i,j})) \left[(W_i \cdot W_j)^2 + 3(W_i \cdot W_j) b_i + 3b_i^2 \right] + N_i \quad (5.42)$$

and N_i is:

$$N_i = (1 - \|W_i\|^2)^2 - 3(1 - \|W_i\|^2)b_i + 3b_i^2. \quad (5.43)$$

At $W_i = 0$:

— $W_i \cdot W_j = 0$ for all j so $\delta(Q_{i,j})$ requires $0 > b_i > 0$, which fails and hence $\delta(Q_{i,j}) = 0$ for all j . Thus, the first sum vanishes and the second sum becomes $3(c - 1)b_i^2$.

— N_i at $\|W_i\|^2 = 0$ becomes $1 - 3b_i + 3b_i^2$.

Thus:

$$H_i^+|_{W_i=0, b_i=b} = 3(c - 1)b^2 + 1 - 3b + 3b^2 = 1 - 3b + 3cb^2. \quad (5.44)$$

We minimize:

$$\frac{dH_i^+}{db} = -3 + 6cb = 0 \Rightarrow b^* = \frac{1}{2c}. \quad (5.45)$$

So at b^* , we have:

$$H_i^+|_{W_i=0, b_i=b^*} = 1 - \frac{3}{4c} \quad (5.46)$$

This calculation doesn't use any dimensional information so its goes through in any other dimension, be it this case or $r = 2, 3$ and so on. The optimal positive vestigial bias is universally $\frac{1}{2c}$, saving $\frac{3}{4c}$ in H per column compared to a standard vestigial.

For the $(1, 1) + \sigma$ -positive $+(c - 2 - \sigma)$ standard configuration:

$$H^* = (c - 2) - \frac{3\sigma}{4c} \quad (5.47)$$

and

$$L^* = \frac{H^*}{3c} = \frac{c - 2}{3c} - \frac{\sigma}{4c^2} . \quad (5.48)$$

This shows that each σ -positive vestigial saves $\frac{1}{4c^2}$ in L . The maximum savings hence are realized when $\sigma = c - 2$ (all vestigials are σ -positive). We can further calculate the loss at this value:

$$L_{min}(c) = \frac{(c - 2)(4c - 3)}{12c^2} . \quad (5.49)$$

We now analyze the Hessian in this case with σ -positive vestigials. Each one contributes a Hessian block in the (W_i, b_i) subspace and we need to expand H_i^+ at $W_i = \epsilon$, $b_i = \frac{1}{2c} + \delta$ (around the critical point $W_i = 0, b_i = \frac{1}{2c}$).

Collecting the terms from H_i^+ and the contributions from the configurations in H_1, H_2 that depend on W_i , we find:

$$H \left(W_i = \epsilon, b_i = \frac{1}{2c} + \delta \right) = H^* + \left(1 + \frac{3}{2c} \right) \epsilon^2 + 3c\delta^2 + O(\epsilon^3) + O(\epsilon\delta) , \quad (5.50)$$

where the cross terms vanish at leading order because the gradient vanishes at the critical point. We find the (W_i, b_i) Hessian block:

$$\nabla^2 H|_{\sigma\text{-positive vestigial}} = \begin{bmatrix} 2 + \frac{3}{c} & 0 \\ 0 & 6c \end{bmatrix} \quad (5.51)$$

Both eigenvalues are strictly positive so there are no flat directions along b_i , in contrast to the standard vestigials. This is a crucial structural difference because the standard vestigials have a flat b_i direction, while σ -positive vestigials have the bias rigidly pinned at $b^* = \frac{1}{2c}$ by the optimization. Moving away from b^* strictly increases H .

The symmetry of the W_i direction also matters:

- from $W_i > 0$ (column joins S_+), $\Delta H = \epsilon^2 + \left(\frac{3}{2c}\right) \epsilon^2 = \left(1 + \frac{3}{2c}\right) \epsilon^2$.
- from $W_i < 0$ (column joins S_-), the same as for S_+ . Symmetric, C^2 .

This means that the σ -positive vestigial column is on a C^2 -smooth wall (not only a C^1 -only wall), thanks to both active sub-groups being non-empty and no directional analysis is needed in this case.

The full Hessian spectrum at $(1, 1) + \sigma$ -positive $+(c - 2 - \sigma)$ -standard) decomposes as block-diagonal with the following:

Spectrum count:

- Positive eigenvalues: $4 + 2\sigma + (c - 2 - \sigma) = c + 2 + \sigma$.
- Zero eigenvalues: $c - 2 - \sigma$ (one per standard vestigial).

Block	Contribution to spectrum
(W_1, b_1) block (active positive)	2 positive eigenvalues $(7 \pm \sqrt{37})$
(W_2, b_2) block (active negative)	2 positive eigenvalues $(7 \pm \sqrt{37})$
Each σ -positive vestigial i	2 positive eigenvalues $(2 + 3/c$ and $6c)$
Each standard vestigial i	1 positive eigenvalue (2) + 1 zero eigenvalue (flat b_i)

- Negative eigenvalues: 0.

$\text{tangent}(\dim) = c - 2 - \sigma$, all coming from the flat standard-vestigial bias directions.

By Morse-Bott, if all non-zero eigenvalues are strictly positive and the kernel matches the tangent space of the critical level, the configuration is minimally singular for every $\sigma \in [0, c - 2]$. The dimension count gives:

$$\lambda(c, \sigma) = \frac{1}{2}(2c - (c - 2 - \sigma)) = \frac{c + 2 + \sigma}{2} . \quad (5.52)$$

So the configuration with σ -positive vestigials has:

$$L(c, \sigma) = \frac{c - 2}{3c} - \frac{\sigma}{4c^2} \quad (5.53)$$

$$\lambda(c, \sigma) = \frac{c + 2 + \sigma}{2} \quad (5.54)$$

and each step in σ trades $\frac{1}{4c^2}$ less loss for $\frac{1}{2}$ more llc, following the rule that increasing complexity (λ), buys lower loss L .

We can index and construct a table in order to see this clearly:

We can now see clearly that as c grows we can keep dropping the loss by increasing σ , with the llc growing linearly. Also, the unique family of minimally-singular points is the $(1, 1) + \sigma$ - positive $+(c - 2 - \sigma)$ - standard family, parameterized by $c \geq 2$ and $\sigma \in [0, c - 2]$. Everything else is a saddle or worse.

5.4.4 Hessian directional analysis

To complete the theoretical analysis we still need to investigate the case of $(1, 0) + (c - 1)$ vestigial configuration where we treat both negative-bias and positive-bias cases to complete the catalog.

The configurations is:

$$(k_+, k_-) = (1, 0) , \quad (5.55)$$

which means we have:

c	σ	L	λ
2	0	0	2
3	0	$1/9 \approx 0.111$	$5/2$
3	1	$1/12 \approx 0.083$	3
4	0	$1/6 \approx 0.167$	3
4	1	$29/192 \approx 0.151$	$7/2$
4	2	$13/96 \approx 0.135$	4
6	0	$2/9 \approx 0.222$	4
6	4	$7/36 \approx 0.194$	6
8	6	$87/512 \approx 0.170$	7
12	10	$97/576 \approx 0.168$	12

Table 5.4: ($r = 1$ ladder of (L, λ) pairs)

— one active positive column: $W_1 = 1, b_1 = 0$

— zero active negative columns

— $c - 1$ vestigial columns

We consider further both possibilities for the vestigial biases: all negative and some/all positive.

The case of negative bias

Where we have $(1, 0) + (c - 1)$ standard vestigials, so:

$$W_1 = 1, \quad b_1 = 0, \quad W_i = 0, \quad b_i = \beta_i < 0 \text{ for } i = 2, \dots, c. \quad (5.56)$$

We compute the specific derivatives to see if it is a critical point or not:

- $\partial H / \partial W_1$: same as the $c = 1$ case (just one active column, no other interactions). Equals zero at $W_1 = 1, b_1 = 0$.
- $\partial H / \partial b_1$: same — zero.
- $\partial H / \partial W_i$ for $i \geq 2$: at $W_i = 0$, both one-sided derivatives are zero (the wedge term in H_1 is $(W_1 W_i)^2$, derivative $2W_1^2 W_i \rightarrow 0$; for $W_i < 0$ side, no wedge active so the derivative vanishes).
- $\partial H / \partial b_i$ for $i \geq 2$: $H_i = 1$ is constant for $b_i \in (-\infty, 0)$, so the derivative vanishes.

This confirms that this is a critical point with:

$$H^* = c - 1 \quad (5.57)$$

$$L^* = \frac{c-1}{3c}, \quad (5.58)$$

which is strictly higher than the $(1, 1) + (c-2)$ - vestigial. Hence this configuration is not a global minimum and we need a directional analysis for each vestigial column (W_i, b_i) .

From the side $W_i > 0$ (column i would join S_+)

In this case we have $W_1 W_i = \epsilon > 0$, $\delta P_{1,i} = 1$ since $b_1 = 0 \geq -\epsilon$ and the contribution from the interaction term is $\frac{(W_1 W_i + b_1)^3}{W_1 W_i} = \epsilon^2$ in H_1 .

Also H_i : with $\|W_i\|^2 = \epsilon^2 > 0$ but $b_i = \beta < 0$ fixed, we need $-\epsilon^2 \leq \beta$ for $\delta(P_i) = 1$. For small ϵ , ϵ^2 is close to 0, but $\beta < 0$ is bounded away from 0. Hence $-\epsilon^2 > \beta$, failing the condition and so $\delta(P_i) = 0$ and $H_i = 1$ stays constant.

For $\delta(P_{i,1})$, the conditions are $W_i W_1 > 0$ and $-W_i W_1 \leq b_i = \beta$, i.e. $-\epsilon \leq \beta$. For $|\beta| > 0$ fails, so find no contribution to H_i . Therefore $\Delta H = \Delta H_1 = \epsilon^2$ from this side.

From the side $W_i < 0$ (column i would join the empty S_-)

Perturbing $W_i = -|\epsilon| < 0$, we find:

— $W_1 W_i = -|\epsilon| < 0$ so $\delta(P_{1,i}) = 0$, which gives no contribution in H_1 .

— For H_i we have the same analysis as above with $\delta(P_i) = 0$, $H_i = 1$ and there is no active negative column to interact with so no interaction term to fire from this side either. Thus $\Delta H = 0$ from this side.

The directional Hessian:

$$\left. \frac{\partial^2 H}{\partial W_i^2} \right|_{W_i=0+} = 2 \quad \left. \frac{\partial^2 H}{\partial W_i^2} \right|_{W_i=0-} = 0. \quad (5.59)$$

The function is C^1 at $W_i = 0$, but not C^2 as the Hessian changes discontinuously.

The set $H = c - 1$ locally near this configuration extends in:

- the b_i direction; $b_i \in (-\infty, 0)$ flat, so this is a full direction.
- $W_i \leq 0$ direction, H is constant on $W_i \leq 0$, so this is a half-line tangent direction, meaning that this configuration sits at the boundary of this half-line, not in the interior.

The configuration hence sits on the boundary defined by the equation $W_i = 0$.

This is not minimally singular in the standard Morse-Bott sense. Hence the level-set here has a boundary and the W_i direction is partly tangent — the $W_i < 0$ half and partly normal — the $W_i > 0$ half.

The configuration is a local minimum because $H \geq c - 1$ and can't roll down in any direction. So it is a critical point with the property of being a local minimum, but with half-space level set. This cannot be treated rigorously as a learning coefficient.

If it were to treat this as a true learning coefficient we would find an erroneous value that does not reflect what actually happens in the neural network:

$$\lambda_{\text{one-side}} = \frac{c+1}{2} . \quad (5.60)$$

The case of positive bias

The set-up is $W_1 = 1$, $b_1 = 0$, and for at least one vestigial i , we try $W_i = 0$, $b_i = \frac{1}{2c}$. We check to see if its a critical point and compute:

$$\frac{\partial H}{\partial W_i}, \text{ at } W_i = 0, \quad b_i = \frac{1}{2c} .$$

The formula for H^+ at small $W_i = \epsilon > 0$ gives:

$$H_i^+ = \sum_{j \neq i} [(W_i W_j)^2 + 3(W_i W_j) b_i + 3b_i^2] + N_i . \quad (5.61)$$

Thus:

$$\frac{\partial H_i^+}{\partial W_i} = \sum_{j \neq i} [2W_i W_j^2 + 3W_j b_i] + \frac{\partial N_i}{\partial W_i} \quad (5.62)$$

and at $W_i = 0$:

$$\left. \frac{\partial H_i^+}{\partial W_i} \right|_{W_i=0} = 3b_i \sum_{j \neq i} W_j . \quad (5.63)$$

Hence for the $(1, 0)$ configuration with $b_i = \frac{1}{2c}$, we have $\sum_{j \neq i} W_j = W_1 + 0 + \dots = 1$ because only W_1 is non-zero among the columns. Therefore:

$$\left. \frac{\partial H_i^+}{\partial W_i} \right|_{W_i=0} = \frac{3}{2c} \neq 0 . \quad (5.64)$$

Because the gradient is non-zero, this configuration with the positive vestigial is not a critical point of H . for comparison, the $(1, 1) + \sigma^-$ positive case had $\sum_{j \neq i} W_j = W_1 + W_2 = 1 + (-1) = 0$ so the gradient vanished.

Concluding this analysis, we found that if the gradient is positive then it pushes W_i in the negative direction in order to decrease H . As W_i becomes negative, the column joins S_- and we end up with a new $(1, 1) + \sigma^-$ positive or standard configuration. This means that the $(1, 0)$ configuration with positive vestigial biases is dynamically unstable and gradient descent would immediately drive one of the vestigial columns into a negative-active state, the system relaxing in the $(1, 1) + (c - 2)$ configuration we already analyzed.

5.5 Experiments which confirm our results for the case $r = 1$

Let us test by numerical experiments if the behavior of the model for the case $r = 1$ agrees with the theoretical results of the last section. For that purpose we built a numerical verification in PyTorch for a ReLU autoencoder with the same specifications as in the theory section, with six input/output neurons and one hidden dimension.

We started from five hand-crafted initial conditions from the following table in order to check if we would find the same functioning:

Table 5.5: *Theoretical predictions*

Configuration	L	λ
(1,1) + standard vest	0.22222	4.0
(1,1) + σ -positive (full)	0.19444	6
(1,0) trap (standard)	0.27778	3.5 [one-sided]
All-same-sign saddle	0.27778	undefined

with an additional first experiment with a random initialization. We used a training loop with SGD with momentum 0.9 and learning rate 0.01 on a dataset of 2048 samples for 3000 epochs, logging the weights snapshots and loss values every 50 SGD steps. We used devinterp’s SGDL-based posterior sample routine in order to estimate the local learning coefficient.

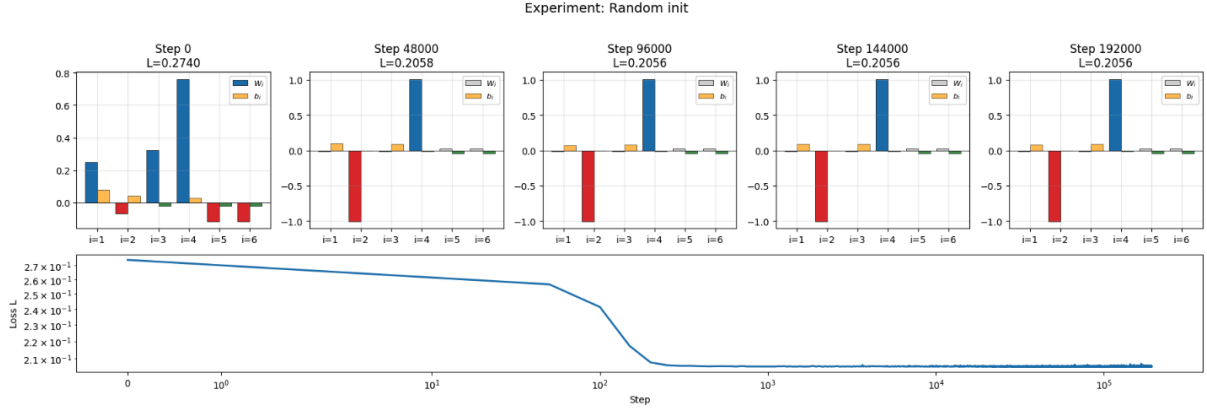
The following results are presented:

5.5.1 Random init

Table 5.6: *Experiment: Random init (Final $L = 0.205616$)*

Index (i)	Initial W	Initial b	Final W	Final b
1	0.24835708	0.07896064	-0.01429462	0.08599225
2	-0.06913215	0.03837174	-1.00484560	-0.00733440
3	0.32384427	-0.02347372	-0.01456960	0.09473342
4	0.76151493	0.02712800	1.01213280	-0.01536356
5	-0.11707669	-0.02317088	0.02374127	-0.04065723
6	-0.11706848	-0.02328649	0.02363842	-0.03975073

With an estimated llc: $\hat{\lambda} \approx 5.426 \pm 0.955$

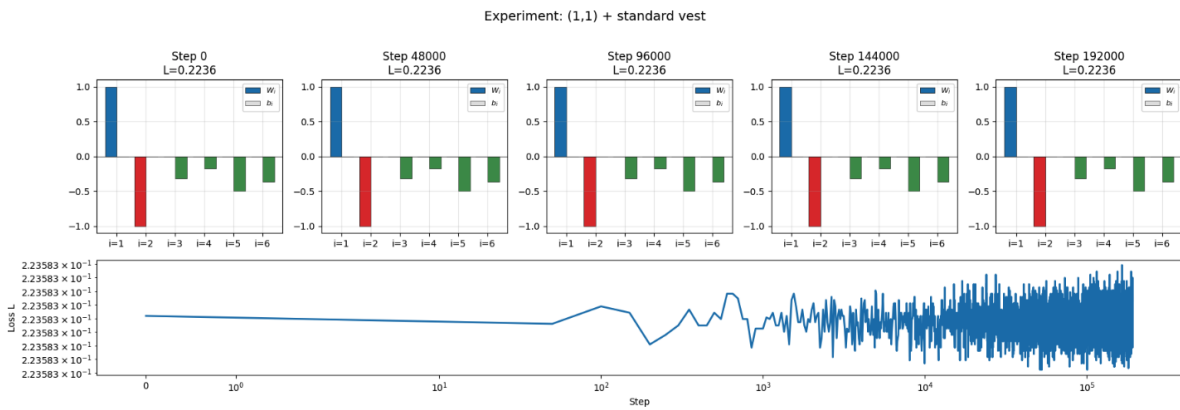


5.5.2 $(1, 1) +$ standard vestigials

Table 5.7: *Experiment: $(1, 1) +$ standard vest (Final $L = 0.223583$)*

Index (i)	Initial W	Initial b	Final W	Final b
1	1.0	0.0	1.0	0.0
2	-1.0	0.0	-1.0	0.0
3	0.0	-0.31234010	0.0	-0.31234010
4	0.0	-0.17585398	0.0	-0.17585398
5	0.0	-0.49994853	0.0	-0.49994853
6	0.0	-0.36395034	0.0	-0.36395034

With an estimated llc: $\hat{\lambda} \approx 3.571 \pm 0.725$



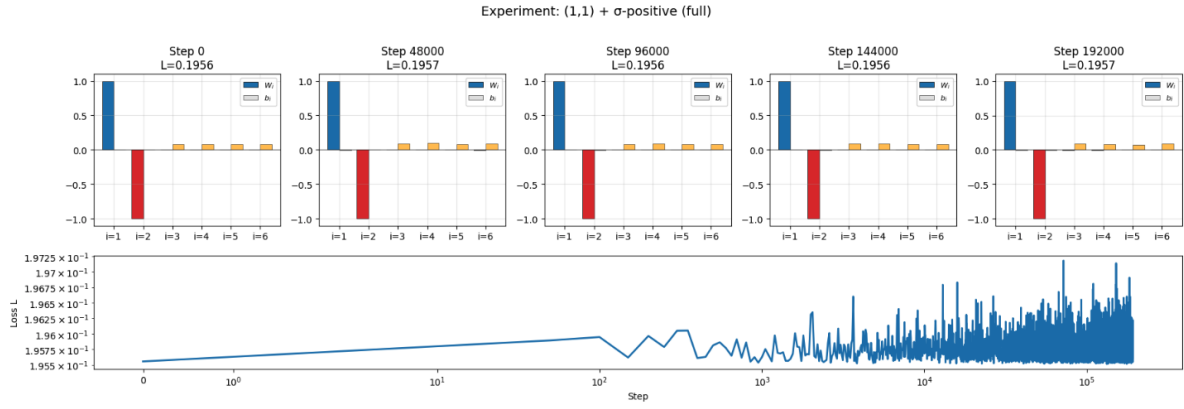
5.5.3 $(1, 1) + \sigma$ - positive vestigials (full)

With an estimated llc: $\hat{\lambda} \approx 7.934 \pm 0.332$

5.5. EXPERIMENTS WHICH CONFIRM OUR RESULTS FOR THE CASE $R = 1.63$

Table 5.8: *Experiment: (1,1) + σ -positive (full) (Final $L = 0.195690$)*

Index (i)	Initial W	Initial b	Final W	Final b
1	1.0	0.0	1.0012797e+00	-0.00223663
2	-1.0	0.0	-1.0013603e+00	-0.00131805
3	0.0	0.08333333	-1.3459936e-03	0.09456643
4	0.0	0.08333333	-1.9398599e-03	0.08519333
5	0.0	0.08333333	-2.6314272e-04	0.07310055
6	0.0	0.08333333	6.3466991e-04	0.09174582

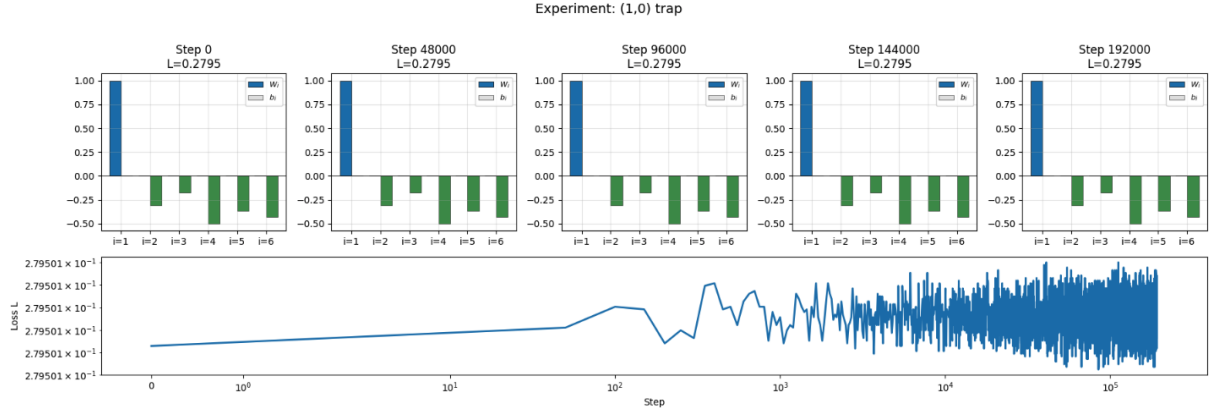


5.5.4 (1,0) trap

Table 5.9: *Experiment: (1,0) trap (Final $L = 0.279501$)*

Index (i)	Initial W	Initial b	Final W	Final b
1	1.0	0.0	1.0	0.0
2	0.0	-0.31234010	0.0	-0.31234010
3	0.0	-0.17585398	0.0	-0.17585398
4	0.0	-0.49994853	0.0	-0.49994853
5	0.0	-0.36395034	0.0	-0.36395034
6	0.0	-0.43395985	0.0	-0.43395984

With an estimated llc: $\hat{\lambda} \approx -6.218 \pm 0.686$

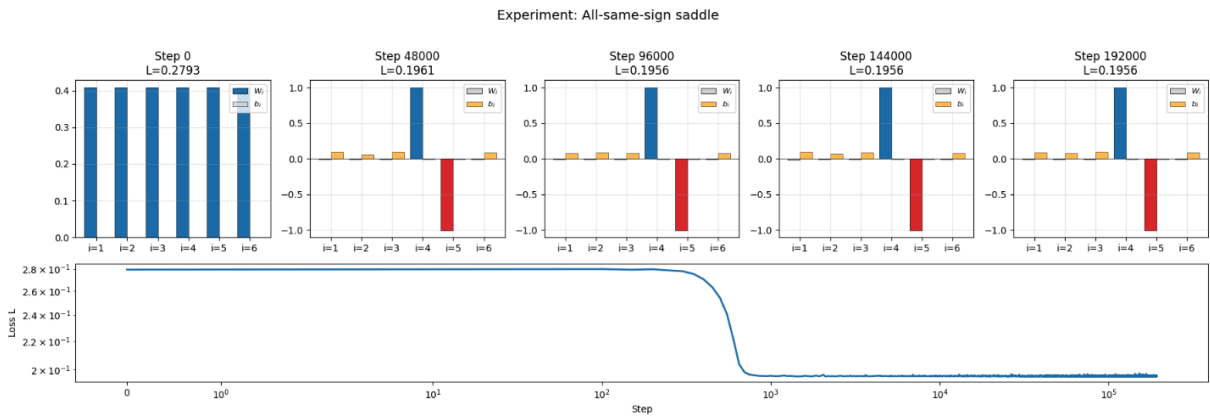


5.5.5 All-same-sign saddle

Table 5.10: *Experiment: All-same-sign saddle (Final $L = 0.195630$)*

Index (i)	Initial W	Initial b	Final W	Final b
1	0.40824829	0.0	-0.01011700	0.08591848
2	0.40824829	0.0	-0.01023294	0.07628817
3	0.40824829	0.0	-0.01064249	0.09466389
4	0.40824829	0.0	1.00500820	-0.00520909
5	0.40824829	0.0	-1.00371610	-0.00639015
6	0.40824829	0.0	-0.01100228	0.09186479

With an estimated llc: $\hat{\lambda} \approx 7.952 \pm 0.607$



5.5.6 Remarks

We found that the overall behavior follows the theory: if we start from a random set of weights, SGD will find a local minimum from the (1, 1) family, with two features encoded along the permitted directions and all four remaining that drift to zero.

On the valid configuration we have an llc which resembles the theoretical prediction or at least is close to it, but for the trap and the saddle we observe the inconclusive behavior that we expected. The trap configuration “fails” the llc prediction in a theoretically expected way, and the saddle shows instability, also which was predicted theoretically. The same phenomena can be seen for the loss, but the main point is that the network behaves as expected.

The Bayesian Antecedent Hypothesis (BAH) discussed in [100] was designed to study superposition and hence transitions between different geometries. In the case $r = 1$ there is no superposition in the sense that the network could represent more features than the number of hidden dimensions — two features can at most be represented without dropping any. The only BAH case that could exist is the trivial transition where the dropped feature sign flips from negative to positive. For a conclusive analysis of the Bayesian Antecedent Hypothesis it is necessary to look at higher dimensions, like the case $r = 2$ considered in [100] or the case $r \geq 3$.

However, our study of the one-dimensional case showed that critical points arise from highly symmetric configurations not only in the case $r = 2$ but also in the case $r = 1$, which suggests that in higher dimensions one should again consider Ansatz based on regular polyhedra. This provides a foundation for extending the analysis to 3-dimensional and higher-dimensional cases.

Conclusions

We explained why there is a rising need for deeper and rigorous study of deep networks, given the increasing risks posed by the capabilities of such systems, which are being deployed and scaled exponentially even though we are far from having full control over their behavior or a full understanding how they operate. We placed such systems in the wider conceptual and historical context of artificial intelligence and explained how they and the current paradigm used for their training arose in the development of that highly interdisciplinary field, drawing from ideas in computational neuroscience and behavioral psychology. We also pointed out the limitations of and dangers inherent in the reinforcement learning paradigm, which treats deep networks as “black boxes”.

We also explained the emerging field of AI safety and its importance for mitigating or avoiding various risks. We then posited that new tools need to be used or created when dealing with such problems, as there is a major difference between the current engineering framework and the hard problems of interpretability which lie at the core of AI safety. Given how these systems are constructed and trained, a natural and powerful conceptual language for analyzing them in this context originates from physics, and especially from statistical mechanics. The field of artificial intelligence has seen much progress during its history by drawing inspiration and methods from other sciences, which makes this connection to physics of no surprise.

Working through the conceptual foundations of machine learning, we built up a language which allows us to understand how a neural network is assembled from its working parts. We then connected Statistical Machine Learning to Statistical Physics, noting deep similarities in their mathematical frameworks which allow us to exchange methods and tools between these subjects. To this end we outlined the basic Bayesian framework of Statistical Learning Theory (which forms the theoretical core of modern machine learning), focusing on its extension to Singular Learning Theory (SLT), which correctly treats models which are singular. Singular models, which are generic in Statistical Learning Theory, are the precise kinds of Bayesian models which are relevant to learning by deep networks as well as to numerous statistical reasoning and model selection problems across natural and social sciences, including various branches of physics.

A key point of the thesis was to summarize Watanabe’s SLT in some mathematical detail. Working through Bayesian statistics, we explained the differences between regular and singular statistical models and showed how one can treat the latter rigorously using Hironaka’s theorem on resolutions of singularities. We then focused on that part of this theory which allows one to define and study phase transitions within the Bayesian learning framework.

To illustrate how SLT becomes tractable despite its highly mathematical character, we chose to study Toy Models of Superposition, which exhibit a plethora of interesting phenomena such as polysemanticity, phase structure and superposition. These are simple ReLU autoencoders operating on high sparsity datasets, an assumption which allows superposition to take place. To analyze such networks within SLT, we followed the methodology of [100], which found that dynamical phase transitions for the class of $r = 2$ models of this type have a Bayesian antecedent. This approach is rigorous and provides a mathematically complete way to study phase transitions in such networks. The authors of [100] also found that the critical points of the potential correspond to emergent geometric structures, which in the case $r = 2$ are represented by regular polygons.

The original contribution of this thesis is our study the $r = 1$ family of TMS models in the SLT framework presented in Section 5, which constitutes a new contribution to their investigation. We showed rigorously that in this case, such models do not exhibit true superposition, essentially because their Bayesian effective dimension is too small — and hence the Bayesian Antecedent Hypothesis is not relevant to this case. We also verified our results using numerical experiments.

Our study of this class of models is a further step towards integrating the fields of physics and artificial intelligence, in order to gain a better understanding of high impact contemporary problems which could provide new insights not only into neural network function and the critically important and fast-evolving subject of AI safety but also in other areas of science where singular Bayesian learning models abound.

References

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Vol. 25. Curran Associates, Inc., 2012, pp. 1097–1105. URL: <https://papers.nips.cc/paper/2012/hash/c3992e4dcd34e4895a65da3528641688-Abstract.html>.
- [2] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. “Sequence to Sequence Learning with Neural Networks”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 27. 2014.
- [3] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017, pp. 5998–6008.
- [4] Tom Brown et al. “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 33. 2020, pp. 1877–1901.
- [5] David Silver et al. “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529.7587 (2016), pp. 484–489. DOI: 10.1038/nature16961.
- [6] David Silver et al. “Mastering the game of Go without human knowledge”. In: *Nature* 550.7676 (2017), pp. 354–359. DOI: 10.1038/nature24270.
- [7] David Silver et al. “A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play”. In: *Science* 362.6419 (2018), pp. 1140–1144. DOI: 10.1126/science.aar6404.
- [8] John Jumper et al. “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* 596.7873 (2021), pp. 583–589. DOI: 10.1038/s41586-021-03819-2.
- [9] Josh Abramson et al. “Accurate structure prediction of biomolecular interactions with AlphaFold 3”. In: *Nature* 630 (2024), pp. 493–500. DOI: 10.1038/s41586-024-07487-w.
- [10] Demis Hassabis. “Accelerating Scientific Discovery with AI”. In: *The Keio Journal of Medicine* 73.4 (2024), pp. 47–47. DOI: 10.2302/kjm.abstract_73_4-1.
- [11] John M. Jumper. *Protein Structure Prediction with AlphaFold: A Machine Learning Approach to Biology*. Nobel Prize Foundation Lecture. Dec. 2024. URL: <https://www.nobelprize.org/prizes/chemistry/2024/jumper/lecture/>.
- [12] Anonymous. “AlphaFold 3: an unprecedented opportunity for fundamental research and drug development”. In: *Precision Clinical Medicine* 8.3 (2025), pbaf015. DOI: 10.1093/pcmedi/pbaf015.

- [13] Warren S. McCulloch and Walter Pitts. “A Logical Calculus of the Ideas Immanent in Nervous Activity”. In: *The Bulletin of Mathematical Biophysics* 5.4 (1943), pp. 115–133.
- [14] Donald O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Wiley, 1949.
- [15] James A. Anderson and Edward Rosenfeld, eds. *Neurocomputing: Foundations of Research*. Cambridge, MA: MIT Press, 1988. ISBN: 9780262010979.
- [16] Jose Miguel Olazaran Rodriguez. “A Historical Sociology of Neural Network Research”. PhD thesis. University of Edinburgh, 1991.
- [17] James A. Anderson and Edward Rosenfeld, eds. *Talking Nets: An Oral History of Neural Networks*. Cambridge, MA: MIT Press, 2000. ISBN: 9780262511117.
- [18] Daniel S. Levine. *From Neuropsychology to Artificial Neural Networks*. Mahwah, NJ: Lawrence Erlbaum Associates, 2000. ISBN: 9780805827958.
- [19] Frank Rosenblatt. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Washington, DC: Spartan Books, 1962.
- [20] Jürgen Schmidhuber. “Deep Learning in Neural Networks: An Overview”. In: *Neural Networks* 61 (2015), pp. 85–117. DOI: 10.1016/j.neunet.2014.09.003.
- [21] Terrence J. Sejnowski. *The Deep Learning Revolution*. Cambridge, MA: MIT Press, 2018. ISBN: 9780262038034.
- [22] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 4th. Pearson, 2020. ISBN: 978-0134610993.
- [23] Daniel A. Roberts, Sho Yaida, and Boris Hanin. *The Principles of Deep Learning Theory*. Cambridge University Press, 2022. DOI: 10.1017/9781009023405.
- [24] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. Adaptive Computation and Machine Learning series. MIT Press, 2016. ISBN: 978-0262035613. URL: <https://www.deeplearningbook.org>.
- [25] Philipp Petersen and Jakob Zech. “Mathematical theory of deep learning”. In: *arXiv preprint arXiv:2407.18384* (2024). DOI: 10.48550/arxiv.2407.18384.
- [26] Arnulf Jentzen, Benno Kuckuck, and Philippe von Wurstemberger. “Mathematical Introduction to Deep Learning: Methods, Implementations, and Theory”. In: *arXiv preprint arXiv:2310.20360* (2025). DOI: 10.48550/arxiv.2310.20360.
- [27] Xiaojing Ye. *Mathematical Foundations of Deep Learning*. Mathematics and Artificial Intelligence Series. Chapman & Hall/CRC, 2026. DOI: 10.48550/arXiv.2603.18387.
- [28] Pedro Domingos. *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*. Basic Books, 2015. ISBN: 978-0465065707.
- [29] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. ISBN: 978-0387310732.
- [30] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. ISBN: 9781107057135.
- [31] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. 2nd. London, UK: The MIT Press, 2018.

- [32] Tong Zhang. *Mathematical Analysis of Machine Learning Algorithms*. Cambridge University Press, 2023. DOI: 10.1017/9781009093057.
- [33] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *Nature* 521.7553 (2015), pp. 436–444. DOI: 10.1038/nature14539. URL: <https://doi.org/10.1038/nature14539>.
- [34] Nestor Maslej et al. *The AI Index 2024 Annual Report*. Tech. rep. Stanford, CA: AI Index Steering Committee, Institute for Human-Centered AI, Stanford University, Apr. 2024.
- [35] Jaime Sevilla et al. “Compute Trends Across Three Eras of Machine Learning”. In: *2022 International Joint Conference on Neural Networks (IJCNN)*. 2022, pp. 1–8. DOI: 10.1109/IJCNN55064.2022.9891914.
- [36] A. Elhanashi. “Generative AI and the Foundation Model Era: A Comprehensive Review”. In: *MDPI Systems* 10.3 (2026), p. 94. DOI: 10.3390/systems10030094.
- [37] Rishi Bommasani et al. “On the Opportunities and Risks of Foundation Models”. In: *arXiv preprint arXiv:2108.07258* (2021). URL: <https://arxiv.org/abs/2108.07258>.
- [38] V. Arunkumar, G. R. Gangadharan, and Rajkumar Buyya. “Agentic Artificial Intelligence (AI): Architectures, Taxonomies, and Evaluation of Large Language Model Agents”. In: *arXiv preprint arXiv:2601.12560* (2026). DOI: 10.48550/arXiv.2601.12560.
- [39] Juhee Kim et al. “The Attack and Defense Landscape of Agentic AI: A Comprehensive Survey”. In: *arXiv preprint arXiv:2603.11088* (2026). Accepted to USENIX Security 2026. DOI: 10.48550/arXiv.2603.11088.
- [40] Wayne Xin Zhao, Kun Zhou, Junyi Li, et al. “A Survey of Large Language Models”. In: *Frontiers of Computer Science* 20 (2026), p. 2012627. DOI: 10.1007/s11704-026-60308-3.
- [41] Humza Naveed et al. “A Comprehensive Overview of Large Language Models”. In: *ACM Transactions on Intelligent Systems and Technology* 16.5 (2025), pp. 1–72. DOI: 10.1145/3744746.
- [42] Shervin Minaee et al. “Large Language Models: A Survey”. In: *arXiv preprint arXiv:2402.06196* (2024).
- [43] Duzhen Zhang et al. *MM-LLMs: Recent Advances in MultiModal Large Language Models*. 2024. DOI: 10.48550/arXiv.2401.13601. arXiv: 2401.13601 [cs.CL].
- [44] Louise Amoore and Volha Piotukh, eds. *Algorithmic Life: Calculative Devices in the Age of Big Data*. London: Routledge, 2016.
- [45] Anthony Elliott. *The Culture of AI: Everyday Life and the Digital Revolution*. London: Routledge, 2019.
- [46] Shoshana Zuboff. *The Age of Surveillance Capitalism*. PublicAffairs, 2019.
- [47] Anthony Elliott. *Making Sense of AI: Our Algorithmic World*. Cambridge: Polity Press, 2022.
- [48] Anthony Elliott. *Algorithmic Intimacy: The Digital Revolution in Personal Relationships*. Cambridge: Polity Press, 2023.

- [49] Raffaele Ciriello, Uri Gal, and Ofir Turel. “Not a Silver Bullet for Loneliness: How Attachment and Age Shape Intimacy with AI Companions”. In: *arXiv preprint arXiv:2602.12476* (2026).
- [50] Jean Baudrillard. *Simulacra and Simulation*. Routledge, 2020, pp. 230–234. DOI: 10.4324/9781003060963-37.
- [51] Joshua Cunningham. “Simulacra on steroids: AI art and the Baudrillardian hyperreal”. In: *Oak Ridge National Laboratory Research Publications* (2025). URL: <https://impact.ornl.gov/en/publications/simulacra-on-steroids-ai-art-and-the-baudrillardian-hyperreal/>.
- [52] Cathryn van Kessel, Jason D. Manriquez, and Kip Kline. “Baudrillard, hyperreality, and the ‘problematic’ of (mis/dis)information in social media”. In: *Theory & Research in Social Education* 53 (2025), pp. 1–23. DOI: 10.1080/00933104.2024.2439302.
- [53] I. R. W. Kusuma. “Hyperreality in the Flexing Culture of Mass Media Communication”. In: *Jurnal Multidisiplin Sahombu* 5.6 (2025), pp. 1908–1918. URL: <https://ejournal.seaninstitute.or.id/index.php/JMS/article/view/7369>.
- [54] Bobby Chesney and Danielle Citron. “Deep fakes: A looming challenge for privacy, democracy, and national security”. In: *California Law Review* 107.6 (2019), pp. 1753–1820. DOI: 10.15779/Z38736M31Z.
- [55] Ben Buchanan et al. *Truth, Lies, and Automation: How Language Models Could Change Disinformation*. Report. Center for Security and Emerging Technology (CSET), 2021. URL: <https://cset.georgetown.edu/publication/truth-lies-and-automation/>.
- [56] Luciano Floridi. “Artificial intelligence, disinformation, and the public sphere”. In: *Philosophy & Technology* 36.1 (2023), pp. 1–12. DOI: 10.1007/s13347-023-00615-w.
- [57] B. F. Skinner. *Science and Human Behavior*. Macmillan, 1953.
- [58] B. F. Skinner. *The Behavior of Organisms*. Appleton-Century, 1938.
- [59] B. F. Skinner. *About Behaviorism*. New York: Alfred A. Knopf, 1974.
- [60] Dario Amodei et al. “Concrete Problems in AI Safety”. In: *arXiv preprint arXiv:1606.06565* (June 2016). DOI: 10.48550/arXiv.1606.06565.
- [61] Dylan Hadfield-Menell et al. “The Off-Switch Game”. In: *arXiv preprint arXiv:1611.08219* (2016).
- [62] Tom Everitt, Gary Lea, and Marcus Hutter. “AGI Safety Literature Review”. In: *IJCAI* (2018), pp. 5441–5449.
- [63] Evan Hubinger et al. “Risks from Learned Optimization in Advanced Machine Learning Systems”. In: *arXiv preprint arXiv:1906.01820* (2019).
- [64] Stuart Russell. *Human Compatible: Artificial Intelligence and the Problem of Control*. New York: Viking, 2019. ISBN: 9780525558613.
- [65] Dan Hendrycks et al. “Unsolved Problems in ML Safety”. In: *arXiv preprint arXiv:2109.13916* (2021).

- [66] Dan Hendrycks, Mantas Mazeika, and Thomas Woodside. “An Overview of Catastrophic AI Risks”. In: *arXiv preprint arXiv:2306.12001* (2023).
- [67] Evan Hubinger et al. “Sleeper Agents: Training Deceptive LLMs that Persist Through Safety Training”. In: *arXiv preprint arXiv:2310.06987* (2023).
- [68] Anthropic. “Alignment Faking in Large Language Models”. In: *arXiv preprint arXiv:2412.14093* (2024).
- [69] Usman Anwar et al. “Foundational Challenges in Assuring Alignment and Safety of Large Language Models”. In: *arXiv preprint arXiv:2404.09932* (2024).
- [70] Huw Roberts et al. “Global AI governance: barriers and pathways forward”. In: *SSRN Electronic Journal* (2023). DOI: 10.2139/ssrn.4588040.
- [71] ITU. *The Annual AI Governance Report 2025: Steering the Future of AI*. Tech. rep. International Telecommunication Union, 2025. URL: <https://www.itu.int/epublications/publication/the-annual-ai-governance-report-2025-steering-the-future-of-ai>.
- [72] OECD. *Governing with Artificial Intelligence: The State of Play and Way Forward in Core Government Functions*. Tech. rep. OECD Publishing, Paris, 2025. URL: <https://doi.org/10.1787/795de142-en>.
- [73] A. Batool, D. Zowghi, and M. Bano. “Responsible artificial intelligence governance: A review and research framework”. In: *AI and Ethics* 5 (2025), pp. 3265–3279. URL: <https://doi.org>.
- [74] Various Authors. “AI Governance in the Context of the EU AI Act”. In: *IEEE Transactions on Artificial Intelligence* (2025). URL: https://www.researchgate.net/publication/394461661_AI_Governance_in_the_Context_of_the_EU_AI_Act.
- [75] X. Wang and F. Xie. “Global artificial intelligence governance research in the digital and intelligent era: advances, trends and countermeasures”. In: *Journal of Knowledge Management* 30.1 (2026), pp. 30–68. DOI: <https://doi.org/10.1108/JKM-01-2025-0006>.
- [76] Yoshua Bengio et al. “International AI Safety Report: First Key Update Capabilities and Risk Implications”. In: *arXiv* 2.6 (2025). DOI: 10.48550/arxiv.2501.17805.
- [77] Yoshua Bengio, Sören Mindermann, Davide Privitera, et al. *International AI Safety Report 2026*. <https://internationalaisafetyreport.org/publication/international-ai-safety-report-2026>. DSIT 2026/001. Feb. 2026. DOI: 10.48550/arXiv.2602.21012. URL: <https://arxiv.org/abs/2602.21012>.
- [78] Sumio Watanabe. *Algebraic geometry and statistical learning theory*. Cambridge University Press, 2009.
- [79] Sumio Watanabe. *Mathematical Theory of Bayesian Statistics*. CRC Press, 2018. DOI: 10.1201/9781315141886.
- [80] Sumio Watanabe. “Review and Prospect of Algebraic Research in Equivalent Framework between Statistical Mechanics and Machine Learning Theory”. In: *arXiv preprint arXiv:2406.10234* (May 2024). DOI: 10.48550/arXiv.2406.10234.

- [81] Edwin T. Jaynes. *Probability Theory: The Logic of Science*. Ed. by G. Larry Bretthorst. Cambridge, UK: Cambridge University Press, 2003. DOI: 10.1017/CB09780511790423.
- [82] Christian P. Robert. *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. 2nd. Springer Texts in Statistics. New York, NY: Springer-Verlag, 2007. ISBN: 9780387715988. DOI: 10.1007/0-387-71599-0.
- [83] Kevin P. Murphy. *Probabilistic Machine Learning: An Introduction*. MIT Press, 2022.
- [84] Andrew Gelman et al. *Bayesian Data Analysis*. 3rd. Chapman & Hall/CRC Texts in Statistical Science. Boca Raton, FL: Chapman and Hall/CRC, 2013. ISBN: 9781439840955. DOI: 10.1201/b16018.
- [85] Gert-Martin Greuel, Christoph Lossen, and Eugenii Shustin. *Introduction to Singularities and Deformations*. Springer Monographs in Mathematics. Berlin, Germany: Springer-Verlag, 2007. ISBN: 9783540283232. DOI: 10.1007/978-3-540-30447-0.
- [86] Vladimir I. Arnold, Sabir M. Gusein-Zade, and Alexander N. Varchenko. *Singularities of Differentiable Maps: Volume I. The Classification of Critical Points, Caustics and Wave Fronts*. Vol. 82. Monographs in Mathematics. Boston, MA: Birkhäuser, 2012. ISBN: 9780817683399. DOI: 10.1007/978-0-8176-8340-5.
- [87] Vladimir I. Arnold, Sabir M. Gusein-Zade, and Alexander N. Varchenko. *Singularities of Differentiable Maps: Volume II. Monodromy and Asymptotics of Integrals*. Vol. 83. Monographs in Mathematics. Boston, MA: Birkhäuser, 2012. ISBN: 9780817683429. DOI: 10.1007/978-0-8176-8343-6.
- [88] Martin Golubitsky and Victor Guillemin. *Stable Mappings and Their Singularities*. Vol. 14. Graduate Texts in Mathematics. New York, NY: Springer-Verlag, 1973. ISBN: 9780387900735. DOI: 10.1007/978-1-4615-7905-2.
- [89] János Kollár. *Lectures on Resolution of Singularities*. Annals of Mathematics Studies 166. Princeton, NJ: Princeton University Press, 2007. ISBN: 9780691129235. DOI: 10.1515/9781400832910.
- [90] Stephen E. Fienberg. “When Did Bayesian Inference Become “Bayesian”?” In: *Bayesian Analysis* 1.1 (2006), pp. 1–40. DOI: 10.1214/06-BA101.
- [91] Wolfgang von der Linden, Volker Dose, and Udo von Toussaint. *Bayesian Probability Theory: Applications in the Physical Sciences*. Cambridge, UK: Cambridge University Press, 2014. DOI: 10.1017/CB09781139165440.
- [92] Udo von Toussaint. “Bayesian inference in physics”. In: *Reviews of Modern Physics* 83.3 (2011), pp. 943–999. DOI: 10.1103/RevModPhys.83.943.
- [93] Michael P. Hobson et al., eds. *Bayesian Methods in Cosmology*. Cambridge, UK: Cambridge University Press, 2010. ISBN: 9780521887670. DOI: 10.1017/CB09780511779633.
- [94] Philip C. Gregory. *Bayesian Logical Data Analysis for the Physical Sciences: A Comparative Approach with Mathematica Support*. Cambridge, UK: Cambridge University Press, 2005. ISBN: 9780521841504. DOI: 10.1017/CB09780511791277.

- [95] Gerhard Bohm and Günter Zech. *Introduction to Statistics and Data Analysis for Physicists*. 2nd revised. Hamburg, Germany: Verlag Deutsches Elektronen-Synchrotron (DESY), 2014. ISBN: 9783935702416. URL: <https://bib-pubdb1.desy.de/record/389738>.
- [96] Manticore Collaboration, Jens Jasche, and Guilhem Lavaux. “Manticore Project I: a digital twin of our cosmic neighbourhood from Bayesian field-level analysis”. In: *Monthly Notices of the Royal Astronomical Society* 540.1 (2025), pp. 716–735. DOI: 10.1093/mnras/stad812.
- [97] Nikoloz Gigiberia. “Bayesian Model Selection with an Application to Cosmology using DES-SN5YR data”. In: *arXiv preprint* (2025). eprint: 2512.09724. URL: <https://arxiv.org/abs/2512.09724>.
- [98] Ezequiel Alvarez, Manuel Szewc, and Santiago Tanco. “Template-Adapted Mixture Models for Unsupervised Signal Fraction Estimation in High-Energy Physics”. In: *arXiv preprint* (2026). eprint: 2605.11605. URL: <https://arxiv.org/abs/2605.11605>.
- [99] Nelson Elhage et al. “Toy Models of Superposition”. In: *Transformer Circuits Thread* (2022). URL: https://transformer-circuits.pub/2022/toy_model/index.html.
- [100] Zhongtian Chen et al. “Dynamical versus Bayesian Phase Transitions in a Toy Model of Superposition”. In: *arXiv preprint arXiv:2310.06301* (2023). DOI: 10.48550/arXiv.2310.06301.
- [101] J. McCarthy et al. *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*. Tech. rep. Dartmouth College, Aug. 1955. URL: <http://jmc.stanford.edu/articles/dartmouth/dartmouth.pdf>.
- [102] A. M. Turing. “Computing Machinery and Intelligence”. In: *Mind* 59.236 (Oct. 1950), pp. 433–460. DOI: 10.1093/mind/LIX.236.433.
- [103] James Lighthill. “Artificial Intelligence: A General Survey”. In: *Artificial Intelligence: A Paper Symposium*. London, UK: Science Research Council, 1973, pp. 1–21.
- [104] Allen Newell, John C. Shaw, and Herbert A. Simon. “Report on a General Problem-Solving Program”. In: *Proceedings of the International Conference on Information Processing*. 1959, pp. 256–264.
- [105] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. “A Formal Basis for the Heuristic Determination of Minimum Cost Paths”. In: *IEEE Transactions on Systems Science and Cybernetics* 4.2 (1968), pp. 100–107. DOI: 10.1109/tssc.1968.300136.
- [106] Noam Chomsky. *Syntactic Structures*. Mouton de Gruyter, 1957.
- [107] Joseph Weizenbaum. “ELIZA—a computer program for the study of natural language communication between man and machine”. In: *Communications of the ACM* 9.1 (1966), pp. 36–45. DOI: 10.1145/365153.365168.
- [108] Kenneth Mark Colby, Sylvia Weber, and Franklin Dennis Hilf. “Artificial paranoia”. In: *Artificial Intelligence* 2.1 (1971), pp. 1–25. DOI: 10.1016/0004-3702(71)90002-6.

- [109] John McCarthy. “Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part I”. In: *Communications of the ACM* 3.4 (1960), pp. 184–195. DOI: 10.1145/367177.367199.
- [110] F. Rosenblatt. “The perceptron: A probabilistic model for information storage and organization in the brain”. In: *Psychological Review* 65.6 (1958), pp. 386–408. DOI: 10.1037/h0042519.
- [111] Marvin Minsky and Seymour A. Papert. *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA: MIT Press, 1969.
- [112] Edward Hance Shortliffe. “Mycin: A Rule-Based Computer Program for Advising Physicians Regarding Antimicrobial Therapy Selection”. Stanford, CA, USA. PhD thesis. Stanford University, 1974.
- [113] Marvin Minsky. *A Framework for Representing Knowledge*. Tech. rep. AIM-306. Cambridge, MA, USA: MIT Artificial Intelligence Laboratory, 1974.
- [114] Randall Davis. “Applications of Meta Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases”. Stanford, CA, USA. PhD thesis. Stanford University, 1976.
- [115] Robert A. Kowalski. “Predicate Logic as Programming Language”. In: *Information Processing* 74 (1974), pp. 569–574.
- [116] Robert Kowalski. “Algorithm = Logic + Control”. In: *Communications of the ACM* 22.7 (1979), pp. 424–436. DOI: 10.1145/359131.359136.
- [117] Alain Colmerauer et al. *Prolog: Manuel de Reference et d’Utilisation*. Tech. rep. Marseille, France: Groupe d’Intelligence Artificielle, Université d’Aix-Marseille, 1973.
- [118] David H. D. Warren. *Implementing Prolog - Compiling Predicate Logic Programs*. Tech. rep. 39/40. Edinburgh, Scotland: Department of Artificial Intelligence, University of Edinburgh, 1977.
- [119] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, 1988.
- [120] Corinna Cortes and Vladimir Vapnik. “Support-Vector Networks”. In: *Machine Learning* 20 (1995), pp. 273–297. DOI: 10.1007/BF00994018.
- [121] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.
- [122] Murray Campbell, A. Joseph Hoane, and Feng-hsiung Hsu. “Deep Blue”. In: *Artificial Intelligence* 134 (2002), pp. 57–83. DOI: 10.1016/s0004-3702(01)00129-1.
- [123] Jia Deng et al. “ImageNet: A Large-Scale Hierarchical Image Database”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.
- [124] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. 2nd ed. Adaptive Computation and Machine Learning series. Cambridge, MA: The MIT Press, 2018.
- [125] John M. Pearce and Geoffrey Hall. *Learning and Memory*. Lawrence Erlbaum, 1992.
- [126] Paul Chance. *Learning and Behavior*. 6th ed. Wadsworth, 2008.

- [127] W. David Pierce and Carl D. Cheney. *Adaptive Behavior and Learning*. Psychology Press, 2013.
- [128] W. David Pierce and Carl D. Cheney. *Behavior Analysis and Learning*. 6th ed. Routledge, 2017.
- [129] William M. Baum. *Understanding Behaviorism: Behavior, Culture, and Evolution*. 3rd ed. Wiley-Blackwell, 2017.
- [130] Erin B. Rasmussen et al. *Behavior Analysis and Learning: A Biobehavioral Approach*. 7th ed. Routledge, 2023.
- [131] Gregory J. Madden, ed. *APA Handbook of Behavior Analysis*. Washington, DC: American Psychological Association, 2013.
- [132] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Belmont, Massachusetts: Athena Scientific, 1996. ISBN: 978-1886529106.
- [133] Dimitri P. Bertsekas. *Abstract Dynamic Programming*. 3rd. Belmont, Massachusetts: Athena Scientific, 2022. ISBN: 978-1886529472.
- [134] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. 4th. Vol. Two-Volume Set. Belmont, Massachusetts: Athena Scientific, 2017. ISBN: 978-1886529083.
- [135] Csaba Szepesvári. *Algorithms for Reinforcement Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2010. DOI: 10.2200/S00268ED1V01Y201005AIM009.
- [136] Clark L. Hull. *Principles of Behavior*. Appleton-Century, 1943.
- [137] Robert R. Bush and Frederick Mosteller. *Stochastic Models for Learning*. Wiley, 1955.
- [138] Robert A. Rescorla and Allan R. Wagner. “A Theory of Pavlovian Conditioning”. In: *Classical Conditioning II*. Appleton-Century-Crofts, 1972.
- [139] Richard J. Herrnstein. “On the Law of Effect”. In: *Journal of the Experimental Analysis of Behavior* 13.2 (1970), pp. 243–266.
- [140] Richard S. Sutton. “Learning to Predict by the Methods of Temporal Differences”. In: *Machine Learning* 3.1 (1988), pp. 9–44.
- [141] Christopher J. C. H. Watkins. “Learning from Delayed Rewards”. PhD thesis. University of Cambridge, 1989.
- [142] Wolfram Schultz, Peter Dayan, and P. Read Montague. “A Neural Substrate of Prediction and Reward”. In: *Science* 275.5306 (1997), pp. 1593–1599.
- [143] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [144] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, et al. “Human-level Control Through Deep Reinforcement Learning”. In: *Nature* 518.7540 (2015), pp. 529–533.
- [145] Jürgen Schmidhuber. “Annotated History of Modern AI and Deep Learning”. In: *arXiv preprint arXiv:2212.11279* (Dec. 2022). DOI: 10.48550/arXiv.2212.11279.
- [146] John J. Hopfield. “Neural networks and physical systems with emergent collective computational abilities”. In: *Proceedings of the National Academy of Sciences* 79.8 (1982), pp. 2554–2558. DOI: 10.1073/pnas.79.8.2554.

- [147] David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. “A learning algorithm for Boltzmann machines”. In: *Cognitive Science* 9.1 (1985), pp. 147–169. DOI: 10.1016/S0364-0213(85)80012-4.
- [148] Edwin T. Jaynes. “Information Theory and Statistical Mechanics”. In: *Physical Review* 106.4 (1957), pp. 620–630. DOI: 10.1103/PhysRev.106.620.
- [149] Edwin T. Jaynes. “Information Theory and Statistical Mechanics. II”. In: *Physical Review* 108.2 (1957), pp. 171–190. DOI: 10.1103/PhysRev.108.171.
- [150] Edwin T. Jaynes. “Prior Probabilities”. In: *IEEE Transactions on Systems Science and Cybernetics* 4.3 (1968), pp. 227–241. DOI: 10.1109/TSSC.1968.300117.
- [151] Edwin T. Jaynes. *E. T. Jaynes: Papers on Probability, Statistics, and Statistical Physics*. Ed. by Roger D. Rosenkrantz. Vol. 158. Synthese Library. Dordrecht, Holland: D. Reidel Publishing Company, 1983. ISBN: 9027714487.
- [152] Alan M. Turing. *Intelligent Machinery, a Heretical Theory*. Lecture given to the ’51 Society at Manchester, broadcast on BBC Third Programme. 1951.
- [153] Norbert Wiener. “Some Moral and Technical Consequences of Automation”. In: *Science* 131.3410 (1960), pp. 1355–1358. DOI: 10.1126/science.131.3410.1355.
- [154] Irving John Good. “Speculations Concerning the First Ultraintelligent Machine”. In: *Advances in Computers*. Ed. by Franz L. Alt and Morris Rubinoff. Vol. 6. Academic Press, 1965, pp. 31–88.
- [155] Nick Bostrom. “Existential Risks: Analyzing Human Extinction Scenarios and Related Hazards”. In: *Journal of Evolution and Technology* 9.1 (2002). URL: <https://www.nickbostrom.com/existential/risks.html>.
- [156] Nick Bostrom and Milan M. Ćirković, eds. *Global Catastrophic Risks*. Oxford: Oxford University Press, 2008.
- [157] Nick Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford: Oxford University Press, 2014. ISBN: 9780198739838.
- [158] Future of Life Institute. *Research Priorities for Robust and Beneficial Artificial Intelligence: An Open Letter*. 2015. URL: <https://futureoflife.org/ai/open-letter-artificial-intelligence/>.
- [159] Future of Life Institute. *Asilomar AI Principles*. 2017. URL: <https://futureoflife.org/open-letter/ai-principles/>.
- [160] Nick Bostrom. “The Vulnerable World Hypothesis”. In: *Global Policy* 10.4 (2019), pp. 455–476.
- [161] Toby Ord. *The Precipice: Existential Risk and the Future of Humanity*. New York: Hachette Books, 2020. ISBN: 9780316484893.
- [162] Center for AI Safety. *Statement on AI Risk*. 2023. URL: <https://www.safe.ai/statement-on-ai-risk>.
- [163] Long Ouyang et al. “Training Language Models to Follow Instructions with Human Feedback”. In: *arXiv preprint arXiv:2203.02155* (2022).
- [164] Paul Christiano et al. “Deep Reinforcement Learning from Human Preferences”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 30. 2017. arXiv: 1706.03741.

- [165] Daniel M. Ziegler et al. *Fine-Tuning Language Models from Human Preferences*. 2019. arXiv: 1909.08593 [cs.CL].
- [166] Richard Ngo, Lawrence Chan, and Sören Mindermann. “The Alignment Problem from a Deep Learning Perspective”. In: *arXiv preprint arXiv:2209.00626* (2022). DOI: 10.48550/arxiv.2209.00626.
- [167] Yuntao Bai et al. “Constitutional AI: Harmlessness from AI Feedback”. In: *arXiv preprint arXiv:2212.08073* (2022).
- [168] Geoffrey Irving, Paul Christiano, and Dario Amodei. “AI Safety via Debate”. In: *arXiv preprint arXiv:1805.00899* (2018).
- [169] Paul Christiano et al. “Supervising Strong Learners by Amplifying Weak Experts”. In: *arXiv preprint arXiv:1810.08575* (2018).
- [170] Collin Burns, Pavel Izmailov, Jan Kirchner, et al. “Weak-to-Strong Generalization: Eliciting Strong Capabilities with Weak Supervision”. In: *arXiv preprint arXiv:2401.04692* (2024).
- [171] Zachary Kenton, Noah Y. Siegel, Janos Kramar, et al. “On Scalable Oversight with Weak LLMs Judging Strong LLMs”. In: *arXiv preprint arXiv:2407.04622* (2024).
- [172] Chris Olah et al. “The Building Blocks of Interpretability”. In: *Distill* (2018).
- [173] Nick Cammarata et al. “Zoom In: An Introduction to Circuits”. In: *Distill* (2020).
- [174] Catherine Olsson et al. “In-context Learning and Induction Heads”. In: *Transformer Circuits Thread* (2022).
- [175] Neel Nanda et al. “Progress Measures for Grokking via Mechanistic Interpretability”. In: *ICLR*. 2023.
- [176] Trenton Bricken, Adly Templeton, et al. “Towards Monosemanticity: Decomposing Language Models With Dictionary Learning”. In: *Transformer Circuits Thread* (2023).
- [177] Department for Science, Innovation and Technology. *The Bletchley Declaration by Countries Attending the AI Safety Summit, 1-2 November 2023*. Nov. 2023. URL: <https://www.gov.uk/government/publications/ai-safety-summit-2023-the-bletchley-declaration>.
- [178] Leonard Bereska and Stratis Gavves. “Mechanistic Interpretability for AI Safety - A Review”. In: *Transactions on Machine Learning Research* (2024). Survey Certification, Expert Certification. ISSN: 2835-8856. URL: <https://openreview.net/forum?id=ePUVetPKu6>.
- [179] Jesse Hoogland et al. *Towards Developmental Interpretability*. LessWrong. July 2023. URL: <https://www.lesswrong.com/posts/TjaeCWvLZtEDAS5Ex/towards-developmental-interpretability>.
- [180] Lee Sharkey et al. “Open Problems in Mechanistic Interpretability”. In: *Transactions on Machine Learning Research* (2025). Survey Certification. ISSN: 2835-8856. URL: <https://openreview.net/forum?id=91H76m9Z94>.
- [181] Leo P. Kadanoff. “Scaling laws for ising models near T_c ”. In: *Physics Physique Fizika* 2 (6 June 1966), pp. 263–272. DOI: 10.1103/PhysicsPhysiqueFizika.2.263.

- [182] Kenneth G. Wilson. “Renormalization Group and Critical Phenomena. I. Renormalization Group and the Kadanoff Scaling Picture”. In: *Phys. Rev. B* 4 (9 Nov. 1971), pp. 3174–3183. DOI: 10.1103/PhysRevB.4.3174.
- [183] Kenneth G. Wilson. “Renormalization Group and Critical Phenomena. II. Phase-Space Cell Analysis of Critical Behavior”. In: *Phys. Rev. B* 4 (9 Nov. 1971), pp. 3184–3205. DOI: 10.1103/PhysRevB.4.3184.
- [184] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge: Cambridge University Press, 2003. ISBN: 9780521642989. URL: <http://www.inference.org.uk/mackay/itila/>.
- [185] Mark J. Schervish. *Theory of Statistics*. Springer Series in Statistics. New York: Springer, 1995. DOI: 10.1007/978-1-4612-4250-5.
- [186] Erich L. Lehmann and George Casella. *Theory of Point Estimation*. 2nd. New York: Springer, 1998. ISBN: 978-0-387-98502-2.
- [187] Aad W. van der Vaart. *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics 3. Cambridge University Press, 1998. DOI: 10.1017/CB09780511802256.
- [188] Lucien Le Cam and Grace L. Yang. *Asymptotics in Statistics: Some Basic Concepts*. 2nd. New York: Springer, 2000. DOI: 10.1007/978-1-4612-1166-2.
- [189] Ole E. Barndorff-Nielsen and David R. Cox. *Inference and Asymptotics*. Monographs on Statistics and Applied Probability 52. London: Chapman & Hall, 1994. DOI: 10.1007/978-1-4899-3210-5.
- [190] Michael J. Kearns and Umesh V. Vazirani. *An Introduction to Computational Learning Theory*. Cambridge, MA: MIT Press, 1994. ISBN: 978-0-262-11193-5.
- [191] Vladimir N. Vapnik. *Statistical Learning Theory*. New York: Wiley, 1998. ISBN: 978-0-471-03003-4.
- [192] Luc Devroye, László Györfi, and Gábor Lugosi. *A Probabilistic Theory of Pattern Recognition*. Stochastic Modelling and Applied Probability 31. New York: Springer, 1996. DOI: 10.1007/978-1-4612-0711-5.
- [193] László Györfi et al. *A Distribution-Free Theory of Nonparametric Regression*. Springer Series in Statistics. New York: Springer, 2002. ISBN: 978-0-387-95441-7. DOI: 10.1007/b97848.
- [194] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning. Cambridge, MA: MIT Press, 2002. ISBN: 978-0-262-19475-4.
- [195] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics 47. Cambridge University Press, 2018. ISBN: 978-1-108-41519-4. DOI: 10.1017/9781108231596.
- [196] Daniel A. Roberts, Sho Yaida, and Boris Hanin. *The Principles of Deep Learning Theory: An Effective Theory Approach to Understanding Neural Networks*. Cambridge University Press, 2022. DOI: <https://doi.org/10.1017/9781009023405>.
- [197] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

- [198] Sebastian Ruder. “An overview of gradient descent optimization algorithms”. In: *arXiv preprint arXiv:1609.04747* (2016).
- [199] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [200] Christopher M. Bishop and Hugh Bishop. *Deep Learning: Foundations and Concepts*. Springer, 2023. DOI: 10.1007/978-3-031-45468-4.
- [201] Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press, 2019. DOI: 10.1017/9781108627771.
- [202] Peter L. Bartlett, Andrea Montanari, and Alexander Rakhlin. *Theoretical Foundations of Deep Learning*. Cambridge University Press, 2024. DOI: 10.1017/9781009025162.
- [203] Asim H. Gazi et al. “Statistical Reinforcement Learning in the Real World: A Survey of Challenges and Future Directions”. In: *arXiv preprint arXiv:2601.15353* (2026).
- [204] Simone Brugiapaglia. “A short tour of operator learning theory: Convergence rates, statistical limits, and open questions”. In: *arXiv preprint arXiv:2603.00819* (2026).
- [205] C. Fiedler. “Statistical Learning Theory for Distributional Classification”. In: *arXiv preprint arXiv:2601.14818* (2026).
- [206] Yuanhe Zhang. “Statistical Learning Theory in Lean 4: Empirical Processes from Scratch”. In: *arXiv preprint arXiv:2602.02285* (2026).
- [207] Susan Wei et al. “Deep Learning Is Singular, and That’s Good”. In: *IEEE Transactions on Neural Networks and Learning Systems* 34.12 (2023), pp. 10473–10486. DOI: 10.1109/TNNLS.2022.3167409.
- [208] Anish Lakkapragada. “Using physics-inspired Singular Learning Theory to understand grokking & other phase transitions in modern neural networks”. In: *arXiv preprint arXiv:2512.00686* (2025). DOI: 10.48550/arXiv.2512.00686. eprint: 2512.00686.
- [209] Sumio Watanabe. “Algebraic geometrical methods for hierarchical learning machines”. In: *Neural Networks* 13.4–5 (2001), pp. 1049–1060. DOI: 10.1016/S0893-6080(00)00069-7.
- [210] Sumio Watanabe. “Algebraic Analysis of Singular Learning Machines”. In: *Neural Networks* 14.8 (2001), pp. 963–985. DOI: 10.1016/S0893-6080(01)00081-3.
- [211] Miki Aoyagi and Sumio Watanabe. “Stochastic Complexities of Reduced Rank Regression in Bayesian Estimation”. In: *Neural Networks* 18.7 (2005), pp. 924–933. DOI: 10.1016/j.neunet.2005.03.010.
- [212] Sumio Watanabe. *The Zeta Function for Learning Theory and Resolution of Singularities*. World Scientific, 2009.
- [213] Sumio Watanabe. “A Widely Applicable Bayesian Information Criterion”. In: *Journal of Machine Learning Research* 14.1 (2013), pp. 867–897. DOI: 10.48550/arxiv.1208.6338.
- [214] Sumio Watanabe. “Information Criteria and Cross Validation for Bayesian Inference in Regular and Singular Cases”. In: *Japanese Journal of Statistics and Data Science* 4 (2021), pp. 1–19.

- [215] Sumio Watanabe. “Recent Advances in Algebraic Geometry and Bayesian Statistics”. In: *Information Geometry* (2022). arXiv: 2211.10049. DOI: 10.1007/s41884-022-00083.
- [216] Sumio Watanabe. “Mathematical Theory of Bayesian Statistics for Unknown Information Source”. In: *Philosophical Transactions of the Royal Society A* (2023). arXiv: 2206.05630. DOI: 10.1098/rsta.2022.0151.
- [217] Edmund Lau et al. “The Local Learning Coefficient: A Singularity-Aware Complexity Measure”. In: *arXiv preprint arXiv:2308.12108* (2023).
- [218] Mathias Drton. “Singular Learning Theory for Factor Analysis”. In: *arXiv preprint arXiv:2511.15419* (2025).
- [219] Maxwell Adam, Zach Furman, and Jesse Hoogland. “The Loss Kernel: A Geometric Probe for Deep Learning Interpretability”. In: *arXiv preprint arXiv:2509.26537* (2025).
- [220] Einar Urdshals et al. “Compressibility Measures Complexity: Minimum Description Length Meets Singular Learning Theory”. In: *arXiv preprint arXiv:2510.12077* (2025).
- [221] Heisuke Hironaka. “Resolution of Singularities of an Algebraic Variety Over a Field of Characteristic Zero: I”. In: *Annals of Mathematics* 79.1 (1964), pp. 109–203. DOI: 10.2307/1970486.
- [222] Herwig Hauser. “The Hironaka theorem on resolution of singularities (or: A proof we can understand)”. In: *Bulletin of the American Mathematical Society* 40.3 (2003), pp. 323–403. DOI: 10.1090/S0273-0979-03-00982-0.
- [223] Trenton Bricken et al. “Towards Monosemanticity: Decomposing Language Models With Dictionary Learning”. In: *Transformer Circuits Thread* (2023). URL: <https://transformer-circuits.pub/2023/monosemantic-features/index.html>.
- [224] Kaarel Hänni et al. “Mathematical Models of Computation in Superposition”. In: *arXiv preprint arXiv:2408.05451* (2024). DOI: 10.48550/arxiv.2408.05451.
- [225] L. Bereska and Efstratios Gavves. “Superposition as Lossy Compression: Measure with Sparse Autoencoders and Connect to Adversarial Vulnerability”. In: *arXiv preprint arXiv:2512.13568* (2025). URL: <https://arxiv.org/abs/2512.13568>.
- [226] D. Chanin et al. “Studying Feature Splitting and Absorption in Sparse Autoencoders”. In: *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS)*. 2025. URL: <https://proceedings.neurips.cc/paper/2025>.
- [227] A. Yildirim. “Superposition Is Not Necessary: A Mechanistic Interpretability Analysis of Transformer Representations for Time Series Forecasting”. In: *arXiv preprint arXiv:2605.05151* (2026). URL: <https://arxiv.org/abs/2605.05151>.
- [228] Shuangshuang Chen and Wei Guo. “Auto-Encoders in Deep Learning—A Review with New Perspectives”. In: *Mathematics* 11.8 (2023), p. 1777. DOI: 10.3390/math11081777.
- [229] Victor Geadah et al. “Sparse-Coding Variational Autoencoders”. In: *Neural Computation* 36 (2024), pp. 2571–2601. DOI: 10.1162/neco_a_01715.
- [230] Yuxiao Li et al. “The Geometry of Concepts: Sparse Autoencoder Feature Structure”. In: *Entropy* 27.4 (2024), p. 344. DOI: 10.3390/e27040344.

- [231] A. Korznikov. “Sanity Checks for Sparse Autoencoders: Do SAEs Beat Random Baselines?” In: *arXiv preprint arXiv:2602.14111* (2026). URL: <https://arxiv.org/abs/2602.14111>.