

Generalisation and expressiveness for over-parameterised neural networks



Eugenio Clerico

Magdalen College

Department of Statistics

University of Oxford

A thesis presented for the degree of

Doctor of Philosophy

Hilary 2023

Statement of Originality

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this or any other university. My personal contributions are as outlined in the authorship forms at the end of each chapter. This dissertation is my own work except as specified in the text, acknowledgements, forms, and papers.

Eugenio Clerico

Hilary 2023

Acknowledgements

First of all, I would like to express my gratitude to my supervisors, George Deligiannidis and Arnaud Doucet. Without their support and guidance, this work would not have been possible. Their mentorship has been invaluable in shaping my research, providing constructive feedback, and offering insightful suggestions.

I would like to thank Benjamin Guedj for giving me the opportunity to work alongside him at UCL. A special thanks to Amitis Shidani and Tyler Farghly, who have not only been great collaborators but have also become close friends. Thanks to Bobby He and Soufiane Hayou for the help and motivation they gave me. I also owe much to all the brilliant colleagues and researchers with whom I had the chance to have valuable and insightful interactions (Judith Rousseau, Patrick Rebeschini, Umut Simsekli, Gergely Neu, Badr-Eddine Chérif-Abdelladif, and many others).

I am grateful to the Engineering and Physical Sciences Research Council (EPSRC) and Magdalen College, Oxford, for allowing me to pursue my DPhil at Oxford. Also, I would like to thank the administrative and IT staff at the Oxford Statistics Department, who were always there to help whenever I needed their support.

This thesis would not have been possible without the encouragement of my friends, both near and far. Jake Fawkes, Shahine Bouabid, Xi Lin, Dan Manela, Anri Asagumo, Jian Qian, Lorenzo Pacchiardi,

Valeria Schellino, Romain Fournier, Yiorgos Kalantzis, Julian Thoeniss, Aizhan Shorman, Florent Bonnet, Kamélia Daudel, Carlo Alfano, Tom Wu, Sam Hall-McMaster, Holly Yeo, and all the other people who shared this journey with me.

I will be eternally grateful to Baba and Adrien for their invaluable friendship and exceptional patience.

Thanks to Marie, Pierre, Filippo and Herbie for their silent support.

Thank you to my family, who has always been there for me: la Mamma, Bruno, Titti, le Nonne, le Zie, gli Zii, i Cugini e chi più ne ha più ne metta. Finally, I want to thank Sekela for suddenly bringing a burst of joy during the final year of my PhD.

Abstract

Over-parameterised modern neural networks owe their success to two fundamental properties: expressive power and generalisation capability. The former refers to the model's ability to fit a large variety of data sets, while the latter enables the network to extrapolate patterns from training examples and apply them to previously unseen data. This thesis addresses a few challenges related to these two key properties.

The fact that over-parameterised networks can fit any data set is not always indicative of their practical expressiveness. This is the object of the first part of this thesis, where we delve into how the input information can get lost when propagating through a deep architecture, and we propose as an easily implementable possible solution the introduction of suitable scaling factors and residual connections.

The second part of this thesis focuses on generalisation. The reason why modern neural networks can generalise well to new data without overfitting, despite being over-parameterised, is an open question that is currently receiving considerable attention in the research community. We explore this subject from information-theoretic and PAC-Bayesian viewpoints, proposing novel learning algorithms and generalisation bounds.

Contents

1	Introduction	1
1.1	Supervised learning framework	2
1.2	Neural networks	4
1.3	Infinite-width limit and Gaussian behaviour	5
1.4	Expressiveness	7
1.5	Generalisation	8
1.5.1	PAC-Bayes	10
1.5.2	Information theoretic bounds	13
1.6	Contributions	15
1.6.1	Stable ResNets	16
1.6.2	Gaussian PAC-Bayes	17
1.6.3	Chained generalisation bounds	20
1.6.4	Deterministic PAC-Bayes under gradient descent	22
2	Stable and expressive ResNets	24
2.1	Preamble	24
2.2	Introduction	25
2.3	Mathematical preliminaries	26
2.3.1	Kernels	26
2.3.2	Gaussian processes	29
2.3.3	Expressiveness and universality	30
2.4	Gaussian limit for neural networks	32
2.4.1	Simple fully-connected neural networks	32
2.4.2	Gaussian limit	33

2.4.3	Expressiveness for finite depth	34
2.4.4	Infinite-depth limit	34
2.4.5	Residual networks	36
2.5	Stable ResNets	37
2.5.1	A toy model	38
2.5.2	Layer- and depth-dependent coefficients	40
2.5.3	Uniform scaling	43
2.5.4	Sequential scaling	44
2.5.5	Expressiveness with no bias	45
2.5.6	Neural tangent kernel	45
2.5.7	A few comments on the empirical results	48
2.6	Omitted proofs	48
2.6.1	Kernels	48
2.6.2	Finite depth	52
2.6.3	Toy model	53
2.6.4	Continuous limit	58
2.6.5	Universality for the uniform scaling	63
2.6.6	Universality for the sequential scaling	65
2.6.7	Universality on the sphere	65
2.7	Statement of authorship	71
3	Gaussian PAC-Bayes	72
3.1	Wide stochastic networks: Gaussian limit and PAC-Bayesian training	73
3.2	Conditionally Gaussian PAC-Bayes	97
3.3	Statements of authorship	116
4	Chained generalisation bounds	118
4.1	Chained Generalisation Bounds	119
4.2	Statement of authorship	165
5	Deterministic PAC-Bayes under gradient descent	166
5.1	Generalisation under gradient descent via deterministic PAC-Bayes	167

CONTENTS

5.2	Statement of authorship	197
6	Discussion	198
6.1	Limitations and open questions	199
6.1.1	Stable and expressive ResNets	199
6.1.2	Gaussian PAC-Bayes	200
6.1.3	Chained generalisation bounds	201
6.1.4	Deterministic PAC-Bayes under gradient descent	202
	References	202

Chapter 1

Introduction

Since the first conception of a programmable computer, people have been curious about the possibility of machines acquiring intelligence (Lovelace, 1842). At present, it has become clear that computers can efficiently perform calculations and tasks practically unsolvable for any human. However, implementing algorithms to execute simple actions that are part of our daily life, such as recognising objects or comprehending spoken sentences, presents a more significant challenge, as it requires expressing our intuitive and subjective understanding in a formal manner. Indeed, early attempts to build a computer, whose knowledge of the world is directly hard-coded in a formal language by a human developer, so far has fallen short of a major success: in order to learn a machine must “acquire [...] knowledge by extracting patterns from raw data” (Goodfellow et al., 2016), a capability known as *machine learning*.

Neural networks have shown the ability to encode knowledge from external environments autonomously. Behind this success is the development of the backpropagation algorithm, which allows to efficiently train multi-layer architectures capable of learning their own representations rather than relying on human-engineered features. Indeed, modern neural networks are structured as the sequential composition of simple parameterised functions, enabling different layers to learn increasingly complex relationships between inputs and outputs. This hierarchical architecture allows the network to extract and combine different types of information from the input, leading to the emergence of more abstract and useful features (LeCun et al., 2015).

For the vast majority of the current state-of-the-art neural networks, the model parameters outnumber by far the training examples available for their tuning. From a mathematical perspective, this translates into a highly complex setting, for which finding rigorous statistical

performance guarantees is still a major open problem (Zhang et al., 2017). Nevertheless, the tremendous empirical success has made multi-layer over-parameterised neural architectures the standard first choice for several learning tasks in various fields, including medicine, email filtering, speech recognition, computer vision, and marketing, among others (LeCun et al., 2015).

Neural networks with millions of parameters can accurately approximate a wide range of functions, a property named *expressiveness* (or *expressivity*). This is usually a desirable quality as it allows the network to learn complex patterns and exhibit great flexibility. However, conventional wisdom suggests that if a model can approximate any function easily, it is likely to overfit the training examples and perform poorly when presented with new data. The ability to extrapolate knowledge from a training data set and apply it effectively to previously unseen instances is called *generalisation*. Despite being over-parameterised, neural networks have demonstrated impressive generalisation capabilities across several tasks. The current lack of a sound theoretical understanding of this phenomenon, and the subsequent difficulty in providing *a priori* statistical performance guarantees, has resulted in the study of generalisation properties of neural networks being an active area of research (Zhang et al., 2017, 2021).

The main focus of this thesis is on the analysis of expressive and generalisation properties of over-parameterised neural networks.

1.1 Supervised learning framework

There are several approaches to machine learning, often divided into the three main categories of supervised learning, unsupervised learning, and reinforcement learning, each suited to different types of tasks and applications. In this context, supervised learning involves training models on labelled data, unsupervised learning deals with finding patterns in unlabelled data, and reinforcement learning entails trial-and-error interactions with an environment. The present thesis will focus on the supervised learning framework.

We consider a space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, whose elements are pairs $z = (x, y)$ called *examples* or *instances*, made of a *feature* $x \in \mathcal{X}$ and its *label* $y \in \mathcal{Y}$. According to the nature of the label space, we speak of classification when \mathcal{Y} is a discrete set, and of regression when the task consists in predicting a continuous quantity. In the simplest scenario there is a ground truth

deterministic map that associate each $x \in \mathcal{X}$ to a unique correct label $y = f^*(x)$. However, we consider the general case where (x, y) are correlated via a probability measure \mathbb{P}_Z on \mathcal{Z} .

The learning task is to build a function $f : \mathcal{X} \rightarrow \mathcal{Y}$ that, given a feature x , can predict its label y . Usually, the algorithm's output is a hypothesis h (belonging to some given space \mathcal{H}), which is understood to parameterise a function $f_h : \mathcal{X} \rightarrow \mathcal{Y}$. In order to gauge how well each hypothesis performs on the examples in \mathcal{Z} , we are given a loss function $\ell : \mathcal{H} \times \mathcal{Z} \rightarrow \mathbb{R}$. We can interpret $\ell(h, z)$ as a measure of how far the model prediction $f_h(x)$ is from the actual label y of x . Typical loss functions are the 0/1 loss (sometimes called misclassification loss) for classification, which is defined as $\ell(h, z) = 1$ if $f_h(x) \neq y$ and 0 otherwise, and the quadratic loss $\ell(h, z) = \|f_h(x) - y\|^2/2$ for regression. Ideally, one would like to find a hypothesis h that minimises the population loss

$$\mathcal{L}_Z(h) = \mathbb{E}_{\mathbb{P}_Z}[\ell(h, Z)]. \quad (1.1)$$

In practice, the law \mathbb{P}_Z is unknown and for our task we are only given a data set $s = \{(x_i, y_i)\}_{i=1}^m \in \mathcal{Z}^m$, consisting of m labelled examples, typically drawn from a distribution \mathbb{P}_S on \mathcal{Z}^m . Often, we will assume that the training data set consists of m independent samples from \mathbb{P}_Z , or from some noisy version of it. As we have only access to a reduced amount of information about \mathbb{P}_Z , we cannot directly evaluate the population loss. A common proxy for this quantity is the empirical loss, obtained by averaging ℓ on the training data set s ,

$$\mathcal{L}_s(h) = \frac{1}{m} \sum_{z \in s} \ell(h, z). \quad (1.2)$$

A typical machine learning strategy is to consider an algorithm that attempts to minimise the empirical loss, namely to find a hypothesis h_* such that, for all $h \in \mathcal{H}$,

$$\mathcal{L}_s(h_*) \leq \mathcal{L}_s(h).$$

From a practical perspective, computing exactly a global minimiser is intractable for most problems of interest, as \mathcal{L}_s is often a highly non-convex function of h . Consequently, the standard approach consists of running computationally efficient optimisation algorithms that return a proxy for h_* .

In the training of neural networks, a typical choice is to opt for iterative first-order methods,

which use the derivative of the optimisation objective \mathcal{C}_s (usually equal to \mathcal{L}_s or a regularised version of it). A vanilla example is the gradient descent algorithm, where one fixes an initial hypothesis h_0 and then updates it iteratively as

$$h_{k+1} = h_k - \eta \nabla \mathcal{C}_s(h_k),$$

where η is a hyper-parameter fixed before the training and called *learning rate*. Under suitable regularity and convexity conditions on \mathcal{C}_s , it is known that the gradient descent converges to the global minimum h_* of the optimisation objective at a fast rate, *i.e.*, polynomial time (Bubeck, 2015). Although these assumptions are known not to hold in many practical settings, in the context of machine learning, first-order methods are often highly effective and typically sufficient to achieve good model performance. Variants of gradient descent, such as stochastic gradient descent (SGD) or AdaGrad, are the *de facto* standard choice for the training of neural networks (Goodfellow et al., 2016).

1.2 Neural networks

In a nutshell, a neural network is a parameterised function with a characteristic structure, consisting in the sequential composition of elementary building blocks. To give a concrete explicit example, we consider the case of a fully-connected feed-forward architecture. The input vector x is mapped to its output $y = F(x)$ by going through a sequence of transformations, typically an affine map composed with a non-linear function. More precisely, let

$$\begin{aligned} F(x) &= u^L(x); \\ u^l(x) &= w^l \phi(u^{l-1}(x)) + b^l; \\ u^0(x) &= w^0 x + b^0. \end{aligned} \tag{1.3}$$

Each u^l is a network's *layer*, a vector whose component are usually referred to as *nodes*. The internal layers $\{u^0, \dots, u^{L-1}\}$ are the so-called *hidden layers*. The *width* of the layer l is defined as the number of nodes of u^l , and often denoted as n_l , while the *depth* is the number of hidden layers (*i.e.*, L). The parameters of the model are $h = \{(w^l, b^l)\}_{l=0}^L$: each w^l is a matrix, named *weight*, while every b^l is a vector, called *bias*. ϕ denotes the activation function

of the network, usually a real mapping that is applied component-wise to all the nodes of the hidden layers. A typical example of activation function is the rectified linear unit (ReLU), defined as $\phi(x) = \max\{0, x\}$.

As a matter of fact, most of neural networks used for practical applications have considerably more complex architectures than the basic one outlined above. They often employ convolutional layers, residual connections, batch normalisation, pooling, as well as other methods and techniques. Nevertheless, at their core, they all rely on a sequential structure, where the input undergoes a series of elementary transformations to produce the output.

As mentioned earlier, the training is usually performed via some gradient-based optimisation method, which requires the practitioner to select an initial configuration h_0 for the parameters. Typically, this is done by drawing h_0 from some simple distribution, as choosing its value deterministically could potentially bias the networks towards unwanted symmetries (Goodfellow et al., 2016). The network’s parameters are then updated in the direction of steepest descent of the optimisation objective.

1.3 Infinite-width limit and Gaussian behaviour

Modern neural networks are made of layers featuring millions of parameters, making an exact analysis of their learning dynamics unfeasible in practice. To address this issue, numerous studies have attempted to identify effective strategies to gain a better understanding of the learning process. One promising approach relies on analysing an approximation of the actual model, which still retains its general behaviour but has the great advantage of being tractable in practice. One example in this sense is the infinite-width asymptotics.

For a simple fully-connected feed-forward network in the form (1.3), the infinite-width limit consists in taking each layer’s width n_l to infinity. However, alternative width definitions allow extending this framework to more general architectures. Different asymptotic regimes for a neural network’s behaviour are observed depending on how the parameters and learning rate scale with the width (see the discussion in Yang and Hu, 2021). This thesis will primarily focus on the Gaussian regime, where the network’s output behaves, at the initialisation, as a Gaussian process.

The first connection between neural networks and Gaussian processes was drawn by Neal

(1995), who considered a fully-connected single-hidden-layer network whose parameters are initialised following independent normal distributions. They showed that, before the training, the output behaves as a centred Gaussian process¹, labelled on the input space. This property allows for the study of the output in terms of a covariance function, which only depends on the network’s activation function and on the variances of weights and biases at the initialisation. In a later work, Lee et al. (2018, 2019) established that a similar conclusion holds for multi-layer fully-connected architectures, where the covariances can be computed recursively. These results were then extended to networks with convolutional layers by Arora et al. (2019), Novak et al. (2019), and Garriga-Alonso et al. (2019), and to more general architectures by Yang (2019b) (introducing the tensor program formalism later expanded and popularised by a series of papers on wide networks by the same author; Yang, 2019a, 2020a; Yang and Littwin, 2021; Yang, 2020b; Yang and Hu, 2021; Yang et al., 2021). From a practical perspective, wide (but finite-size) networks exhibit an almost Gaussian behaviour. This fact was shown empirically by Lee et al. (2019, 2020) and various subsequent papers, while several works give finite size corrections to the asymptotic limit (e.g., Antognini, 2019; Basteri and Trevisan, 2022).

In a nutshell, the output and each hidden node of an infinitely wide neural network behave like independent Gaussian processes at the initialisation. The idea behind this result is quite intuitive. To see things more explicitly, we consider a simple fully-connected feed-forward architecture (1.3), where the parameters are initialised as

$$w_{ij}^l \sim \mathcal{N}(0, \sigma_w^2/n_{l-1}); \quad b_i^l \sim \mathcal{N}(0, \sigma_b^2),$$

with σ_w and σ_b denoting some positive standard deviations, n_{l-1} the input dimension. In this way, each node in the layer l is a sum of independent identically distributed random variables coming from the previous layer. If the number of nodes in the layer $l - 1$ tends to infinity, by the central limit theorem we get that, for each fixed input, the output has a Gaussian law.

However, as pointed out by Matthews et al. (2018), things are more delicate if we want to conclude on the Gaussianity of the whole process. In particular, one shall pay attention to the order with which the widths of different layers are taken to infinity, and to which type

¹A Gaussian process labelled on \mathcal{X} is a family of random variables $\{T_x : x \in \mathcal{X}\}$ such that for any finite set $\{x_1, \dots, x_n\} \subset \mathcal{X}$, $(T_{x_1}, \dots, T_{x_n})$ follows a multinormal distribution. Interestingly, the law of a Gaussian process is entirely determined by its mean function $M(x) = \mathbb{E}[T_x]$ and its covariance function $Q(x, x') = \mathbb{E}[T_x T_{x'}] - M(x)M(x')$.

of convergence holds. Then, one has to deal with the fact that the input space might be infinite, in general even uncountable, a question often neglected in the literature. In [Matthews et al. \(2018\)](#), the authors formally proved that, when all the widths of the layers $l \in [1 : L]$ are brought to infinity simultaneously, the network’s output converges in law to a centred Gaussian process, provided that the input space is countable. Further rigorous analysis can be found in [Yang \(2019b,a\)](#), where the convergence is proven using the tensor programs formalism for a wide range of architectures but under the hypothesis of a finite input set. Finally, in later works, [Hanin \(2021\)](#) and [Bracale et al. \(2021\)](#) provided convergence proofs holding with uncountable input spaces.

Having a good understanding of wide networks at the initialisation can be helpful in practice. For instance, it can help set the optimal hyper-parameters for the training. For example, a careful choice of σ_b and σ_w (the standard deviation for the parameters at the initialisation) can lead to faster and better training for deep architectures ([Schoenholz et al., 2017](#); [Hayou et al., 2019b](#)). More recently, [Yang et al. \(2021\)](#) built on the infinite-width asymptotic analysis to propose a hyper-parameter tuning strategy for large models.

As a final remark, when the learning rate is small enough (scales inverse-proportionally with the width), an infinitely wide neural network obeys simple linear dynamics in functional space. Indeed, the training evolution follows a kernel gradient descent governed by the so-called neural tangent kernel (NTK). This result was first established by [Jacot et al. \(2018\)](#) and later extended, refined, and generalised (*e.g.*, [Lee et al., 2019](#); [Yang and Littwin, 2021](#)).

1.4 Expressiveness

Intuitively, a large number of parameters leads to high expressive power. It has been known since the late ‘80s that shallow neural networks (*i.e.*, with arbitrary width and bounded depth) are universal approximators ([Cybenko, 1989](#); [Funahashi, 1989](#); [Hornik et al., 1989](#)). This means that for a variety of relevant functional classes \mathcal{F} and for any required level of precision, a wide enough shallow neural network can approximate all functions in \mathcal{F} . Similar results were later shown for architectures with fixed width and unrestricted depth ([Lu et al., 2017](#); [Hanin and Sellke, 2018](#); [Kidger and Lyons, 2020](#)).

Recently, a popular research trend has been to compare the expressive power of deep versus

shallow neural networks. For instance, [Montufar et al. \(2014\)](#) and [Poole et al. \(2016\)](#) showed that a network’s expressiveness scales exponentially with the depth, whilst several works have pointed out that it is possible to construct functions easily approximable by deep networks but requiring far much more complex shallow architectures to achieve the same accuracy (*e.g.*, [Telgarsky, 2015, 2016](#); [Cohen et al., 2016](#); [Eldan and Shamir, 2016](#); [Safran and Shamir, 2017](#); [Rolnick and Tegmark, 2018](#)).

Although the above universality results imply that a deep and large network can almost perfectly fit any data set, practical issues can arise when training very deep architectures. This suggests that mere universal approximation results are not always enough to gauge the actual expressiveness that a model can practically achieve (see [Section 1.6.1](#) and [Chapter 2](#) for further discussion). In a seminal work, [Schoenholz et al. \(2017\)](#) pointed out that the network-equivalent Gaussian process of a wide architecture becomes trivial at the initialisation, as the number of layers approaches infinity. In simple terms, each randomly initialised layer adds noise to the input, and for a large number of layers this can cause the output to forget about its input, becoming a constant or mere noise. This loss of input information during the forward propagation is exponential in the depth and brings about trainability issues ([Schoenholz et al., 2017](#)). Interestingly, a careful choice of the variances of the parameters at the initialisation ([Schoenholz et al., 2017](#); [Hayou et al., 2019b](#)), or conditions of orthogonality on the initial weight matrices ([Hu et al., 2020](#)), allow the input information to propagate deeper through the network and lead to better training results in practice.

1.5 Generalisation

In machine learning, *generalisation* refers to a model’s ability to perform accurately on new and unseen examples after training on a limited data set. Until recently, conventional wisdom had it that there must be a trade-off between expressiveness and generalisation, since a model with too high expressive power could easily overfit noise in the training data, thereby failing to capture the underlying meaningful patterns. Nonetheless, modern neural networks have challenged this traditional belief: in spite of their over-parameterisation, they can generalise extremely well. This fact has made apparent that a more comprehensive theoretical framework is needed, as our current mathematical tools are insufficient to provide a satisfying understanding of this

phenomenon.

Generalisation bounds are a fundamental tool that aims at quantifying the gap between the expected performance of the algorithm on new data (the population loss \mathcal{L}_Z defined in (1.1)) and the one on the training examples (the empirical loss \mathcal{L}_s given by (1.2)). These bounds are useful in practice, since they can help assess the effectiveness of machine learning algorithms, compare different models, and guide the hyper-parameters tuning. Furthermore, they play a crucial role in understanding theoretical underpinnings of machine learning, such as the interplay between model complexity, input distribution, and sample size.

The first generalisation bounds relied on measures of the hypothesis space's complexity, such as VC dimension or Rademacher complexity (Vapnik, 2000; Bousquet et al., 2004). However, these bounds are algorithm-independent by nature (*i.e.*, they hold even for the worst algorithm on the given hypothesis space), a fact that often makes them unsuitable for dealing with over-parameterised neural networks (Shalev-Shwartz and Ben-David, 2014; Zhang et al., 2017). To address this issue, recent approaches aim at providing algorithm-dependent generalisation bounds, which differ from the traditional methods in that they focus on the hypotheses likely to be selected instead of considering the complexity of the entire hypothesis space \mathcal{H} .

A number of these generalisation bounds rely on the concept of algorithmic stability, which builds on the intuition that a hypothesis less dependent on the specific data set used for the training is less susceptible to overfitting and will therefore generalise better. This thesis will mainly focus on two approaches inspired by this principle, namely information-theoretic and PAC-Bayesian bounds, which will be introduced in greater detail in the upcoming sections. Several other methods that leverage the perspective of stability consider how much the output of a learning algorithm is affected if a single element of the training data set is modified or removed. This idea was put forth in the late '70s by Devroye and Wagner (1979) for leave-one-out cross-validation and has more recently led to the concept of *uniform stability*, proposed by Bousquet et al. (2004). Uniform stability has been shown to hold for a variety of problems and algorithms (including support vector machines for regression and classification), and can lead to generalisation bounds in high probability on the draw of the training data set (Bousquet et al., 2004; Feldman and Vondrak, 2019; Bousquet et al., 2020). Several recent studies have further refined this technique to establish bounds for iterative algorithms (*e.g.*,

Elisseeff et al., 2005; Hardt et al., 2016; Charles and Papailiopoulos, 2018).

It is worth mentioning that the generalisation literature offers several approaches other than stability, which can also be used in combination with the aforementioned methods. Among these, we recall bounds based on margin arguments, where not only the predictor’s error but also its confidence is taken into consideration (*e.g.*, Novikoff, 1962; Cortes and Vapnik, 1995; Bartlett et al., 1998; Langford and Shawe-Taylor, 2002; Bartlett et al., 2005, 2017). Additionally, there are bounds that leverage the chaining method (Dudley, 1967; Talagrand, 1996), a high-dimensional probability tool that has recently been applied to the PAC-Bayesian (Audibert and Bousquet, 2004) and information-theoretic (Asadi et al., 2018; Asadi and Abbe, 2020; Clerico et al., 2022b) frameworks (see Section 1.6.3 and Chapter 4 for more details). Lastly, it is worth mentioning the use of local complexity measures for the hypothesis space, such as the local Rademacher complexity (Bartlett et al., 2005).

As a final remark, we shall note that the literature on generalisation bounds includes other results that could not find space in the above overview, such as lower bounds on the generalisation gap or excess risk bounds that estimate the discrepancy in performance between the algorithm’s output and the optimal hypothesis h_* that minimises the population loss. However, a discussion of these topics is beyond the scope of the present thesis and has hence been omitted.

1.5.1 PAC-Bayes

The PAC-Bayesian theory provides a framework to establish generalisation guarantees for randomised predictors. This entails considering an extension of the setting introduced in Section 1.1, dealing with stochastic algorithms that, given a training set s , produce a distribution $\mathbb{P}_{H|s}$ on \mathcal{H} , rather than a hypothesis h . The PAC-Bayesian bounds are upper bounds on the expected population loss, $\mathbb{E}_{\mathbb{P}_{H|S}}[\mathcal{L}_{\mathcal{Z}}(H)]$, that hold with high probability on S , the randomly drawn training data set. The guiding principle is the idea of stability introduced earlier: a distribution $\mathbb{P}_{H|S}$ that does not depend too heavily on the specific training set S is likely to yield good generalisation. In practice, this ‘dependence’ is often measured via the relative

entropy,

$$\text{KL}(\mathbb{P}_{H|S} \parallel \mathbb{P}_H^*) = \begin{cases} \mathbb{E}_{\mathbb{P}_{H|S}} \left[\log \frac{d\mathbb{P}_{H|S}}{d\mathbb{P}_H^*} \right] & \text{if } d\mathbb{P}_{H|S} \ll d\mathbb{P}_H^*; \\ +\infty & \text{otherwise,} \end{cases}$$

between an arbitrary distribution \mathbb{P}_H^* (whose choice cannot rely on S) and the final distribution $\mathbb{P}_{H|S}$. \mathbb{P}_H^* and $\mathbb{P}_{H|S}$ are usually referred to as *prior* and *posterior* respectively, in analogy with the Bayesian literature (see [Germain et al. \(2016\)](#) for a discussion on connections and differences between the PAC-Bayesian framework and Bayesian inference).

The PAC-Bayesian theory originated in the seminal work of [Shawe-Taylor and Williamson \(1997\)](#) and [McAllester \(1998\)](#). Its bounds are usually classified as either empirical or oracle. In this thesis, we will concentrate solely on the former ones. We refer to [Alquier \(2021\)](#) for an overview of the latter ones, which were introduced by [Catoni \(2003, 2004, 2007\)](#) and compare the randomised predictor of interest to the hypothesis minimising the population loss.

It is often possible to explicitly evaluate the empirical PAC-Bayesian bounds, as long the expected empirical loss $\mathbb{E}_{\mathbb{P}_{H|S}}[\mathcal{L}_S(H)]$ and the relative entropy between prior and posterior can be computed. The earliest of these results were derived by [McAllester \(1998, 1999\)](#) and typically scale as $O\left(\sqrt{\text{KL}(\mathbb{P}_{H|S} \parallel \mathbb{P}_H^*)/m}\right)$, where m is the dimension of the training set. Later works ([Langford and Seeger, 2001](#); [McAllester, 2003a](#); [Maurer, 2004](#); [Catoni, 2007](#); [Tolstikhin and Seldin, 2013](#); [Mhammedi et al., 2019](#)) proposed and discussed new bounds that scale at the “fast rate” $\frac{1}{m}$ for small enough empirical losses. A more formal theory and a broader framework, as well as novel bounds and techniques, were developed in [Catoni \(2004, 2007\)](#), which after almost two decades still remain major references in the field, while [Germain et al. \(2009\)](#) provided a “general recipe” to establish PAC-Bayesian bounds. The framework was also applied to Gaussian process classification by [Seeger \(2002\)](#) and coupled with margin techniques (*e.g.*, [Langford and Shawe-Taylor, 2002](#); [McAllester, 2003b](#); [Neyshabur et al., 2018](#); [Biggs and Guedj, 2022](#); [Biggs et al., 2022](#)), chaining methods ([Audibert and Bousquet, 2004](#); [Clerico et al., 2022b](#)), and sparsity argument (*e.g.*, [Dalalyan and Tsybakov, 2008](#); [Alquier and Lounici, 2010](#); [Alquier and Biau, 2013](#); [Guedj and Alquier, 2013](#); [Chérif-Abdellatif, 2020](#)). Most of the results mentioned so far apply to bounded losses for classification. Among the works dealing with regression and unbounded losses, we mention [Audibert \(2004\)](#), [Alquier \(2008\)](#), [Audibert and Catoni \(2011\)](#), [Rivasplata et al. \(2020\)](#), and [Haddouche et al. \(2021\)](#). We refer to [Guedj \(2019\)](#) and [Alquier \(2021\)](#) for recent surveys of the PAC-Bayesian literature.

Recently, [Dziugaite and Roy \(2017\)](#) prompted a new surge of interest in the PAC-Bayesian framework. They implemented the training of a stochastic neural network, whose parameters follow independent Gaussian variables with learnable means and variances, by using a PAC-Bayesian bound as the optimisation objective. Actually, the idea of training an algorithm by optimising a PAC-Bayesian bound over a simple family of posterior (variational PAC-Bayes) was not a novelty (*e.g.*, [Langford and Caruana, 2002](#); [Germain et al., 2009](#); [Alquier et al., 2016](#)). However, [Dziugaite and Roy \(2017\)](#) were the first to successfully apply it to obtain non-vacuous bounds for over-parameterised multi-layer architectures. Follow-up works have further improved these empirical bounds, with experiments on image recognition tasks, such as MNIST, CIFAR, and ImageNet (*e.g.*, [Zhou et al., 2018](#); [Pérez-Ortiz et al., 2021b,a](#); [Clerico et al., 2022a, 2023a](#)). Chapter 3 focuses on some algorithms and techniques for this kind of PAC-Bayesian training.

The tightest bounds from these experiments were achieved with the use of data-dependent priors, which can be obtained by splitting the data set into two: one part is used to choose a prior, and the other to tune the posterior and evaluate the bounds (a method originally proposed by [Seeger \(2002\)](#) and [Parrado-Hernández et al. \(2012\)](#)). Other possible approaches to make the prior data-dependent are the localisation technique ([Catoni, 2007](#)) and differential privacy ([Dziugaite and Roy, 2018](#)). We refer to [Dziugaite et al. \(2021\)](#) for further discussion on the role of data in the choice of the prior.

As a last remark, we mention disintegrated PAC-Bayesian bounds, holding in high probability for a realisation of a predictor drawn from the posterior distribution. They will be the object of Chapter 5 and will be introduced in Section 1.6.4.

A simple concrete example To give a more tangible idea of what a PAC-Bayesian bound looks like, we give an example that can be seen as the result of a refined union bound. Assume that ℓ is bounded in $[0, 1]$, fix $\delta \in (0, 1)$ and $\lambda > 0$, and let $\mathbb{P}_S = \mathbb{P}_Z^{\otimes m}$. By Hoeffding’s inequality (see, *e.g.*, [Boucheron et al., 2013](#)), for any fixed $h \in \mathcal{H}$ (chosen independently of S)

$$\mathbb{P}_S \left(\mathcal{L}_Z(h) \leq \mathcal{L}_S(h) + \frac{1}{\sqrt{8m}} \left(\lambda + \frac{\log(1/\delta)}{\lambda} \right) \right) \geq 1 - \delta.$$

For a finite hypothesis space $\mathcal{H} = \{h_1 \dots h_N\}$, via a simple union argument, we can obtain a generalisation bound that holds uniformly for all the hypotheses, so that we can choose the

best h according to the training set. We have

$$\mathbb{P}_S \left(\forall h, \quad \mathcal{L}_Z(h) \leq \mathcal{L}_S(h) + \frac{1}{\sqrt{8m}} \left(\lambda + \frac{\log N + \log(1/\delta)}{\lambda} \right) \right) \geq 1 - \delta.$$

Refining the argument, one can extend the bound to the countable case (Bousquet et al., 2004).

This involves defining a prior measure $\hat{\mathbb{P}}_H$ on \mathcal{H} , and yields

$$\mathbb{P}_S \left(\forall h, \quad \mathcal{L}_Z(h) \leq \mathcal{L}_S(h) + \frac{1}{\sqrt{8m}} \left(\lambda + \frac{\log(1/\hat{\mathbb{P}}_H(h)) + \log(1/\delta)}{\lambda} \right) \right) \geq 1 - \delta.$$

Note that this generalises the previous result: when \mathcal{H} is finite, the uniform distribution ($\hat{\mathbb{P}}_H(h) = 1/N$ for all h) leads to a factor $\log N$.

In the uncountable case, things get somewhat trickier, as most hypotheses will likely have a zero probability mass under $\hat{\mathbb{P}}_H$. However, we can find something meaningful by considering a stochastic algorithm that picks a distribution $\mathbb{P}_{H|S}$ instead of selecting a single hypothesis. We can then upper bound the expected value of the population loss as

$$\mathbb{P}_S \left(\forall \mathbb{P}_{H|S}, \quad \mathbb{E}_{\mathbb{P}_{H|S}}[\mathcal{L}_Z(H) - \mathcal{L}_S(H)] \leq \frac{1}{\sqrt{8m}} \left(\lambda + \frac{\text{KL}(\mathbb{P}_{H|S} \| \hat{\mathbb{P}}_H) + \log(1/\delta)}{\lambda} \right) \right) \geq 1 - \delta,$$

which was originally derived by Catoni (2003) and whose proof's main ingredients are the sub-Gaussianity² of the bounded loss ℓ and the variational formulation of the relative entropy (see, e.g., Boucheron et al., 2013). From this last result, we recover the bound for the countable case by looking only at posteriors in the form $\mathbb{P}_{H|S} = \delta_h$ for $h \in \mathcal{H}$.

1.5.2 Information theoretic bounds

The PAC-Bayesian bounds introduced in the previous section hold with high probability on the draw of the training data set. An alternative approach to control the generalisation is to look at guarantees in expectation under \mathbb{P}_S . This is the case for several information-theoretic generalisation bounds. As the name suggests, this approach considers the generalisation problem from an information-theory standpoint, viewing the algorithm as a noisy channel connecting the training data set and the hypotheses. Inspired by the concept of stability introduced earlier, these bounds build on the intuition that an excessive amount of shared

²A random variable U is σ -sub-Gaussian if $\log \mathbb{E}[e^{\lambda U}] \leq \lambda \mathbb{E}[U] + \frac{\lambda^2 \sigma^2}{2}$, for all $\lambda > 0$.

information between h and s will likely result in poor generalisation.

Mathematically, the idea is quite simple: one needs to compare the joint law of H and S (which we denote as $\mathbb{P}_{H,S}$) with the product of the two marginals, $\mathbb{P}_{H \otimes S} = \mathbb{P}_H \otimes \mathbb{P}_S$.³ The starting point is to notice that we can write the expected generalisation gap as

$$\mathcal{G} = \mathbb{E}_{\mathbb{P}_{H,S}}[\mathcal{L}_Z(H) - \mathcal{L}_S(H)] = \mathbb{E}_{\mathbb{P}_{H,S}}[\mathcal{L}_S(H)] - \mathbb{E}_{\mathbb{P}_{H \otimes S}}[\mathcal{L}_S(H)].$$

Under suitable regularity conditions on the loss, we can control this object in terms of how much $\mathbb{P}_{H,S}$ and $\mathbb{P}_{H \otimes S}$ are ‘far apart’.

The first bound of this kind, due to a 2016’s version of [Russo and Zou \(2019\)](#) and then re-derived by [Xu and Raginsky \(2017\)](#) in a more general setting, is in the form

$$|\mathcal{G}| \leq \sqrt{\frac{I(H; S)}{2m}}, \quad (1.4)$$

where I denotes the mutual information $I(H; S) = \text{KL}(\mathbb{P}_{H,S} \| \mathbb{P}_{H \otimes S})$, and holds under the assumption of sub-Gaussianity⁴ of $\ell(h, Z)$, under the examples’ distribution \mathbb{P}_Z for all h .

Several variants of (1.4) have been proposed in the literature, often to overcome the fact that the mutual information bound is infinite for a deterministic algorithm. On the one hand, different measures of the shared information between H and S have been proposed. For instance, under suitable Lipschitz regularity for the loss, we can obtain bounds based on the Wasserstein distance ([Lopez and Jog, 2018](#); [Wang et al., 2019](#)), while conditions weaker than sub-Gaussianity yield bounds building on Rényi α -divergences ([Esposito et al., 2021](#)). On the other hand, several strategies allow for sharpening the mutual information bound. In this line, we find bounds based on conditioning on a supersample ([Steinke and Zakynthinou, 2020](#)) or considering the mutual information between h and each training input ([Bu et al., 2019](#)). Another possible approach makes use of the chaining technique, an idea first proposed by [Asadi et al. \(2018\)](#) and that we further developed in [Clerico et al. \(2022b\)](#) (see Section 1.6.3 and Chapter 4 for more details). Connections with the PAC-Bayesian literature have also been explored, for instance in [Grunwald et al. \(2021\)](#). Finally, [Lugosi and Neu \(2022\)](#) uses arguments from convex analysis and online learning to build novel information-theoretic

³Here, the marginal \mathbb{P}_H is defined as the measure satisfying $\mathbb{P}_H(A) = \mathbb{E}_{\mathbb{P}_S}[\mathbb{P}_{H|S}(A)]$, for any event A on \mathcal{H} .

⁴See footnote 2.

results.

We remark that these information-theoretic bounds are mostly theoretical tools, as they are hard to evaluate in practice and involve expectations under the unknown training data distribution \mathbb{P}_S . Nevertheless, they provide natural intuition on the mechanism of the learning process, and, as a result, they represent a very active research area. Moreover, recent works have built on them to derive empirical bounds for specific algorithms, such as Langevin dynamics, stochastic gradient Langevin dynamics, and stochastic gradient descent (Bu et al., 2019; Haghifam et al., 2020; Rodríguez-Gálvez et al., 2020; Neu et al., 2021).

A notable line of research around information-theoretic bounds concerns whether they can characterise min-max rates⁵ for specific learning problems. For binary classification, although the vanilla mutual information bound from Russo and Zou (2019) fails in this task for specific hypothesis classes (Bassily et al., 2018), the conditionally mutual information bound by Steinke and Zakyntinou (2020) achieves this goal (Haghifam et al., 2021). However, for problems outside binary classification, Haghifam et al. (2023) recently showed that these bounds cannot obtain min-max rates for stochastic convex optimisation. This fact has raised concerns on whether information-theoretic and PAC-Bayesian approaches are suitable tools for the study of generalisation for over-parameterised models and if refining and combining these approaches with other techniques would overcome this issue.

1.6 Contributions

Overall, this thesis contributes to various aspects of deep learning and statistical learning theory, introducing novel methods to enhance model performance and deepen our understanding of the behaviour of over-parameterised neural networks.

The work presented in the chapters that follow led to five separate papers. The work on the expressive power of infinitely wide and deep residual architectures that we published in Hayou et al. (2021) is revisited in Chapter 2, with some omissions and supplementary results included to emphasise my contributions. On the other hand, Chapters 3 to 5 report four papers in their entirety. Chapter 3 includes Clerico et al. (2022a) and Clerico et al. (2023a), both discussing

⁵The min-max rate is a classical way to measure the ‘learnability’ of a problem. It characterises how fast the population loss goes to zero, with the number of available training examples, for the best learning algorithm in the worst-case scenario (*i.e.*, when the target is the most difficult to learn given the available data).

PAC-Bayesian training techniques that exploit the Gaussianity of a network’s output. Chapter 4 (Clerico et al., 2022b) establishes a framework to derive information-theoretic bounds based on the chaining technique. Chapter 5 (Clerico et al., 2023b) proposes disintegrated PAC-Bayesian bounds that leverage the network’s randomness at the initialisation of models trained via descent algorithms. Finally, Chapter 6 summarises this thesis’s key results and outlines possible avenues for future research.

1.6.1 Stable ResNets (Hayou et al., 2021)

The equivalent Gaussian process of a wide network’s output becomes trivial in the limit of an infinite number of layers (Schoenholz et al., 2017). From an information-theoretic perspective, the network is a noisy channel connecting input and output, the noise due to the random initialisation sums up layer by layer and, as the depth of the architecture diverges, causes the final output to forget its original input completely. This phenomenon manifests as follows: the output is either a random constant or entirely chaotic (*i.e.*, discontinuous almost everywhere), a dichotomous behaviour often referred to as order-chaos phase transition. A consequence of the output’s triviality at the initialisation is that networks in the infinite-depth regime cannot be trained, at least with gradient-based algorithms: heuristically, a network which is ‘blind’ about its input cannot learn anything. More rigorously, as the number of layers goes to infinity, the output’s gradient (with respect to the parameters) can be shown to vanish or explode almost surely.

A reasonable attempt to prevent the input information’s loss is the introduction of skip connections⁶ that map the identity from one layer to the next one. An architecture with this property is usually called a residual network (ResNet). However, a vanilla implementation of this approach translates into the output’s explosion as the depth grows.

In our work, we propose to overcome this issue by adding suitable layer- and depth-dependent scaling factors to the architecture. As a simple example, consider a ResNet

⁶This means modifying (1.3) as $u^l = u^{l-1} + w^l \phi(u^{l-1}) + b^l$.

$F : \mathbb{R}^p \rightarrow \mathbb{R}^q$, defined as

$$\begin{aligned} F(x) &= u^L(x); \\ u^l(x) &= u^{l-1}(x) + \frac{1}{\sqrt{L}} \left(w^l \phi(u^{l-1}(x)) + b^l \right); \\ u^0(x) &= w^0 x + b^0, \end{aligned}$$

where, at initialisation, $w^l \sim \mathcal{N}(0, \sigma_w^2/n_{l-1})$ and $b^l \sim \mathcal{N}(0, \sigma_b^2)$, with $n_{-1} = p$.

In the infinite-width limit, each layer is associated with a centred Gaussian process, labelled on the input space \mathcal{X} . The covariance functions can be evaluated by recursion and, in the case of a ReLU network (*i.e.*, with $\phi : u \mapsto \max\{0, u\}$), we can get explicit expressions for them.

In our work, we focus on studying the expressiveness of the output at the initialisation, in the limit of infinite depth ($L \rightarrow \infty$). Using functional analytic tools (reproducing kernel Hilbert spaces and compact self-adjoint operators theory), we show that skip connections and scaling factor together allows one to obtain a non-exploding and fully expressive output, in the sense that the limiting Gaussian process can approximate any function in L^2 , with non-zero probability. We then also show that the neural tangent kernel, describing the network’s evolution during the training, is fully expressive, meaning that the model can fit any dataset.

Lastly, our empirical results, over a range of image recognition tasks, show that introducing the scaling that we propose can improve the performance of the trained networks.

1.6.2 Gaussian PAC-Bayes (Clerico et al., 2022a, 2023b)

Several recent works (*e.g.*, Dziugaite and Roy, 2017; Pérez-Ortiz et al., 2021a,b) have obtained generalisation guarantees for stochastic classifiers trained by optimising a PAC-Bayesian bound via gradient descent methods. Often, the model considered is a neural network, whose parameters H are independently normally distributed, and the training consists in tuning their means \mathbf{m} and standard deviations \mathfrak{s} (let us call them *hyper-parameters*). The PAC-Bayesian prior distribution is defined by the values of \mathbf{m} and \mathfrak{s} at the initialisation, while the final values of these hyper-parameters characterise the posterior.

On the one hand, in this simple setting, evaluating the relative entropy between prior and posterior is particularly easy, as we are just considering multivariate normal distributions. On the other hand, computing the expected empirical loss can be extremely tricky since, in general,

the law of the output can be very complex. Consequently, the standard practice consists of sampling, at each training iteration, a realisation of the parameters H . Then, in the training objective the expected empirical loss is replaced with its realisation. However, this approach cannot work if we want to use the 0/1 loss⁷, which is constant almost everywhere and hence yields a null gradient for each realisation of the parameters. As a practical solution, a surrogate loss is often used in the training. However, it is worth noting that if we could compute the expected empirical loss exactly, we would not need to rely on any surrogate loss, as the 0/1 loss would bring a non-zero gradient. To have a more concrete intuition of this, we can imagine what happens if we try to go down the stairs via gradient descent: a single point (*i.e.*, single realisation) on a horizontal step cannot decide the correct direction to go down, but a continuous distribution of points has a global view of the stairs.

Wide stochastic networks (Clerico et al., 2023a) We establish that a shallow stochastic network with a single hidden layer has a Gaussian output in the infinite-width limit. More precisely, consider a sequence $\{F^{(n)}\}_{n \in \mathbb{N}}$ of stochastic networks $\mathbb{R}^p \rightarrow \mathbb{R}^q$ with increasing width n , defined by

$$F^{(n)}(x) = \frac{1}{\sqrt{n}} W_1^{(n)} \phi \left(\frac{1}{\sqrt{p}} W_0^{(n)} x \right),$$

where ϕ is the activation function, and $W_1^{(n)}$ (respectively $W_0^{(n)}$) is a $q \times n$ (respectively $n \times p$) weight matrix, whose components are independent normal random variables with means $\mathbf{m}_1^{(n)}$ (respectively $\mathbf{m}_0^{(n)}$) and standard deviations $\mathfrak{s}_1^{(n)}$ (respectively $\mathfrak{s}_0^{(n)}$). If we initialise all components of $\mathbf{m}_1^{(n)}$ and $\mathbf{m}_0^{(n)}$ independently from a standard normal $\mathcal{N}(0, 1)$, and all the components of $\mathfrak{s}_1^{(n)}$ and $\mathfrak{s}_0^{(n)}$ at 1, then we have that for each input x

$$F^{(n)}(x) \rightarrow \mathcal{N}(M(x), Q(x))$$

as $n \rightarrow \infty$, in probability with respect to the random initialisation and in distribution with respect to the intrinsic stochasticity of the network. The q -vector $M(x)$ and the $(q \times q)$ -matrix $Q(x)$ are limits of analytic functions of the hyper-parameters.

Under the assumption of a lazy training regime, namely when the hyper-parameters do not move too much from their initial values, the Gaussian limit holds true even during the

⁷ $\ell(h, z) = 1$ if $f_h(x) \neq y$ and 0 otherwise.

training. For instance, this occurs when the relative entropy between the prior and posterior distributions (defined by the initial and final values of the hyper-parameters, respectively) stays of order $O(1)$ as $n \rightarrow \infty$. In particular, if the optimisation objective penalises the growth of the relative entropy (as is the case with a PAC-Bayesian bound), then the network’s output will be Gaussian throughout the whole training procedure.

Interestingly, we can exploit this Gaussian limit to perform a PAC-Bayesian training on the 0/1 loss directly, without the need of a surrogate loss. Indeed, knowing the exact output distribution allows one to compute both the expected empirical loss on the training data set and its gradient with respect to the hyper-parameters. However, when we use a finite-width network, the output’s law is only approximately Gaussian. In our work, we propose to train a wide network as if its law was exactly Gaussian and analytically compute the gradient descent step in this approximation. Then, once the training is complete, sampling multiple realisations of the parameters we can obtain a probabilistic upper bound on the expected empirical loss, and hence a rigorous generalisation guarantee. Our experiments on the MNIST dataset show that this approach leads to tighter bounds than the standard PAC-Bayesian methods proposed in previous studies (Dziugaite and Roy, 2017; Pérez-Ortiz et al., 2021a).

Conditionally Gaussian PAC-Bayes (Clerico et al., 2022a) A significant limitation of the approach that we just presented is that it only ensures the output’s Gaussianity for a shallow neural network with a single hidden layer. This is because the convergence result is based on a central limit theorem that assumes the independence of the hidden nodes, which is no longer true for a network with multiple hidden layers.

However, if the last layer’s parameters are normally distributed, the output is still Gaussian when conditioned over all the hidden parameters (an idea also exploited in Biggs and Guedj, 2021). Leveraging this, we propose to sample the hidden layers’ parameters at each iteration. Conditioned on this realisation, the output is Gaussian, and we can perform a gradient step avoiding the need for a surrogate loss.

This conditionally Gaussian PAC-Bayesian training approach works for rather general multi-layer stochastic architectures. In our experiments with MNIST and CIFAR10 with convolutional networks, our method outperformed the previous state-of-the-art PAC-Bayesian training results reported in Pérez-Ortiz et al. (2021a).

1.6.3 Chained generalisation bounds (Clerico et al., 2022b)

Many of the information-theoretic bounds from the literature do not consider the dependencies between different hypotheses. Indeed, it is often the case that if two hypotheses h and h' are “close” (according to some notion of distance on \mathcal{H}), then they lead to similar generalisation gaps (*i.e.*, $\mathcal{L}_{\mathcal{Z}}(h) - \mathcal{L}_s(h) \simeq \mathcal{L}_{\mathcal{Z}}(h') - \mathcal{L}_s(h')$), for most of the training data sets. In order to take into account this property, Asadi et al. (2018) introduced a mutual information bound that builds on the chaining technique.

Given a sequence of finer and finer discretisations \mathcal{H}_k of the space \mathcal{H} , we write as h_k the projection of a hypothesis h on \mathcal{H}_k . The main idea behind the chaining is to rewrite the quantity of interest (here, the generalisation gap) as a telescopic sum. Formally, we have

$$\mathcal{L}_{\mathcal{Z}}(h) - \mathcal{L}_s(h) = \mathcal{L}_{\mathcal{Z}}(h_0) - \mathcal{L}_s(h_0) + \sum_{k=1}^{\infty} [(\mathcal{L}_{\mathcal{Z}}(h_k) - \mathcal{L}_s(h_k)) - (\mathcal{L}_{\mathcal{Z}}(h_{k-1}) - \mathcal{L}_s(h_{k-1}))],$$

where one shall make a rigorous sense of the convergence. Asadi et al. (2018) established a mutual information bound in the form

$$\mathcal{G} = \mathbb{E}_{\mathbb{P}_{H,S}}[\mathcal{L}_{\mathcal{Z}}(H) - \mathcal{L}_S(H)] \leq \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2I(H_k; S)/m},$$

where each ε_k is the “length-scale” of the discretisation \mathcal{H}_k . The main assumption for this result is that the differences $\ell(Z, h) - \ell(Z, h')$ are $\|h - h'\|$ -sub-Gaussian⁸ under \mathbb{P}_Z . In this way, we consider the dependencies between different hypotheses. Note that each term in the right hand side contains the mutual information between a discretised hypothesis and the training data set. These quantities are finite (as long as \mathcal{H} is bounded), so this chained mutual information bound can be finite even for a deterministic algorithm.

In our work, we investigate whether it is possible to obtain chained bounds that are not based on the mutual information. First, we build an abstract framework encompassing several information-theoretic bounds from the literature. Then, we show that a chained counterpart corresponds to each of these results.

In order to make things more explicit, let us introduce a notion of function regularity. Given a generic mapping \mathfrak{D} that takes two probability distributions \mathbb{P} and $\hat{\mathbb{P}}$ on some space

⁸See footnote 2.

\mathcal{A} , and returns a positive real value (which we might interpret as a measure of dissimilarity between \mathbb{P} and $\hat{\mathbb{P}}$), we say that $f : \mathcal{A} \rightarrow \mathbb{R}$ is \mathfrak{D} -regular with respect to \mathbb{P} if, for any $\hat{\mathbb{P}}$, we have

$$|\mathbb{E}_{\mathbb{P}}[f] - \mathbb{E}_{\hat{\mathbb{P}}}[f]| \leq \mathfrak{D}(\mathbb{P}, \hat{\mathbb{P}}).$$

If one can establish that the loss follows this definition of regularity, it is almost straightforward to deduce a generalisation bound.

Theorem 1. *Assume that $s \rightarrow \mathcal{L}_s(h)$ is \mathfrak{D} -regular with respect to \mathbb{P}_S , $\forall h \in \mathcal{H}$. Then*

$$|\mathcal{G}| = |\mathbb{E}_{\mathbb{P}_{H \otimes S}}[\mathcal{L}_S(H)] - \mathbb{E}_{\mathbb{P}_{H,S}}[\mathcal{L}_S(H)]| \leq \mathbb{E}_{\mathbb{P}_H}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|H})].$$

Different choices of \mathfrak{D} allow for recovering several results from the literature. For instance, the mutual information bound from [Russo and Zou \(2019\)](#) follows from the fact that a sub-Gaussian function is \mathfrak{D} -regular with $\mathfrak{D}(\mathbb{P}, \hat{\mathbb{P}}) = \sqrt{2\text{KL}(\hat{\mathbb{P}}\|\mathbb{P})}$, while we can obtain a Wasserstein bound from [Lopez and Jog \(2018\)](#) by using that a Lipschitz function is \mathfrak{D} -regular when \mathfrak{D} is the 1-Wasserstein distance.

In order to obtain chained bounds, we need to control how much the loss changes when we consider two distinct hypotheses. Under suitable conditions, it is enough to ask for a regularity assumption on the loss's gradient. Here, we extend the previous definition of \mathfrak{D} -regularity by saying that a q -vector-valued function f is \mathfrak{D} -regular with respect to \mathbb{P} if, for every $v \in \mathbb{R}^q$ with $\|v\| = 1$, $v \cdot f$ is \mathfrak{D} -regular with respect to \mathbb{P} .

Theorem 2 (Informal). *Let \mathcal{H} be convex and compact, and let ℓ be regular enough. If $s \mapsto \nabla_h \mathcal{L}_s(h)$ is \mathfrak{D} -regular with respect to \mathbb{P}_S , $\forall h \in \mathcal{H}$, then*

$$|\mathcal{G}| = |\mathbb{E}_{\mathbb{P}_{H \otimes S}}[\mathcal{L}_S(H)] - \mathbb{E}_{\mathbb{P}_{H,S}}[\mathcal{L}_S(H)]| \leq \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_H}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|H_k})].$$

In particular, this result shows that lifting the regularity assumption from the loss to its gradient makes it possible to obtain a chained bound. This way, we can recover the chained mutual information bound ([Asadi et al., 2018](#)) and establish the novel chained counterpart of the Wasserstein bound from [Lopez and Jog \(2018\)](#).

The other main point addressed in our work is whether the chained bounds are tighter than their unchained counterparts. There is no general answer to this question. However,

although the assumptions required for the chained bounds are somewhat stronger, there are situations where the chained bound performs much better. For instance, with the help of a few simple toy models, we show that this is the case when \mathbb{P}_H is highly concentrated in a tiny region of \mathcal{H} .

1.6.4 Deterministic PAC-Bayes under gradient descent (Clerico et al., 2023b)

One peculiarity of the PAC-Bayesian bounds is that they require the model to be stochastic, which is not the standard setting in many cases. However, if we consider the case of the so-called “deterministic” neural networks, the initialisation is usually drawn from some simple distribution. In our work, we propose a PAC-Bayesian bound that only requires the model’s initialisation to be random.

Most PAC-Bayesian bounds in the literature are in expectation (under the posterior). However, there are also disintegrated results, which hold with high probability with respect to the joint law of the training data set and the parameters (under the posterior). A simple example is the following bound (Alquier, 2021), which requires the loss to be bounded in $[0, 1]$ and holds with probability higher than $1 - \delta$ on the pair (S, H) :

$$\mathcal{L}_{\mathcal{Z}}(H) \leq \mathcal{L}_S(H) + \frac{\lambda}{8m} + \frac{\log \frac{d\mathbb{P}_{H|S}}{d\mathbb{P}_H^*}(H) + \log \frac{1}{\delta}}{\lambda}.$$

Here, \mathbb{P}_H^* is a data-agnostic prior, while $\mathbb{P}_{H|S}$ any arbitrary posterior distribution absolutely continuous with respect to \mathbb{P}_H^* .

Let us consider a training performed by continuous time gradient descent:

$$\partial_t H_t = -\nabla \mathcal{C}(H_t), \tag{1.5}$$

where \mathcal{C} denotes some objective function (e.g., \mathcal{L}_S). This ODE defines a flow in the hypothesis space and, in particular, induces a family of probability measures \mathbb{P}_{H_t} , obtained as the push-forwards of \mathbb{P}_{H_0} under this flow: for all t we have $H_t \sim \mathbb{P}_{H_t}$. Now, fix a time horizon $T > 0$ and consider the algorithm that outputs H_T . If we say that $H_0 \sim \mathbb{P}_H^*$ (i.e., the prior coincides with the initialisation \mathbb{P}_{H_0}), we get that the posterior $\mathbb{P}_{H|S}$ coincide with \mathbb{P}_{H_T} , and sampling

H_T from it can be obtained by first sampling H_0 from the prior \mathbb{P}_{H_0} , and then following the ODE dynamics up to time T .

For simplicity, let us assume that, for all t , \mathbb{P}_{H_t} admits a Lebesgue density ρ_t . Fixed T , we have that with probability higher than $1 - \delta$ on $(S, H_0) \sim \mathbb{P}_S \otimes \mathbb{P}_{H_0}$,

$$\mathcal{L}_{\mathcal{Z}}(H_T) \leq \mathcal{L}_S(H_T) + \frac{\lambda}{8m} + \frac{\log \frac{\rho_T(H_T)}{\rho_0(H_T)} + \log \frac{1}{\delta}}{\lambda}.$$

From the continuity equation $\partial_t \rho_t(h) = \nabla \cdot (\rho_t(h) \nabla \mathcal{C}(h))$, with a little algebra we obtain that $\partial_t(\log \rho_t(H_t)) = \Delta \mathcal{C}(H_t)$, with Δ denoting the Laplacian. This allows us to rewrite

$$\log \frac{\rho_T(H_T)}{\rho_0(H_T)} = \log \frac{\rho_0(H_0)}{\rho_0(H_T)} + \int_0^T \Delta \mathcal{C}(H_t) dt$$

and hence obtain the bound

$$\mathcal{L}_{\mathcal{Z}}(H_T) \leq \mathcal{L}_S(H_T) + \frac{\lambda}{8m} + \frac{\log \frac{\rho_0(H_0)}{\rho_0(H_T)} + \int_0^T \Delta \mathcal{C}(H_t) dt + \log \frac{1}{\delta}}{\lambda}. \quad (1.6)$$

Usually, the term $\log \frac{\rho_0(H_0)}{\rho_0(H_T)}$ can be easily evaluated, as it solely requires the knowledge of the initialisation density ρ_0 . Moreover, in principle, the integral is computable as well, as we only need the value of a local quantity (the Laplacian) along the trajectory followed by H_t during the training.

The bound (1.6) is a simple instance of the more general results that our approach can produce for the continuous-time dynamics. In our work, we also show that under suitable smoothness conditions on the training objective, it is possible to obtain similar bounds for discrete-time dynamics, such as the often-used stochastic gradient descent algorithm. We discuss these and other results, and compare them with others from the literature.

Chapter 2

Stable and expressive ResNets

2.1 Preamble

Most of this chapter constitutes the core of [Hayou et al. \(2021\)](#), a paper accepted at AISTATS 2021, which is the outcome of a collaboration with two other graduate students in the Statistics department: Soufiane Hayou and Bobby He. I contributed to the theoretical side of the project, in particular, mostly on the study of expressiveness and universality.

My major contributions to [Hayou et al. \(2021\)](#) are probably Proposition A1 (corresponding to Proposition 2 here), allowing for the extension of the universality results from the sphere to a generic compact $K \in \mathbb{R}^p$, the whole discussion about the uniform scaling and its continuous limit, and the rigorous formulation of the duality universality/expressiveness (see Lemma 3 in [Hayou et al. \(2021\)](#) and its stronger versions, Lemma 2 and Lemma 4, in this chapter).

In order to emphasise my actual contribution to the project, some of the results of [Hayou et al. \(2021\)](#) are omitted: the study of a PAC-Bayesian bound for Gaussian process kernel regression and the discussion on the explosion and stabilisation of the gradient, which were entirely due to Soufiane. Also, the empirical results of [Hayou et al. \(2021\)](#) are only quickly summarised at the end of this chapter, as they were mainly performed by Bobby, with solid help from Soufiane.

In contrast, the findings presented in Section 2.5.1 are unpublished and were obtained before our collaboration began. At that time, I was considering the expressiveness of uniformly rescaled infinitely deep ResNets, by examining the network-equivalent Gaussian process with elementary tools from functional analysis, without resorting to reproducing kernels.

Additionally, the discussion on the equivalence between expressiveness and universality, as well as some other expressiveness results presented here, are extensions of what was reported in [Hayou et al. \(2021\)](#).

The details of my contribution to the paper can also be found at the end of this chapter.

2.2 Introduction

A popular approach for studying over-parameterised networks involves focusing on the limit of infinite width. For fully-connected multi-layer architectures, this means taking the number of nodes in each hidden layer to infinity. Although this is impossible to achieve in practice, infinitely wide networks possess many interesting properties that can help grasp the complex behaviour of large (but finite-size) networks. One remarkable fact is that, at the initialisation, infinitely wide networks behave like Gaussian processes, a result first established by [Neal \(1995\)](#) for the 1-layer case and later extended to multi-layer architectures ([Lee et al., 2018](#); [Matthews et al., 2018](#); [Lee et al., 2019](#)). From a theoretical standpoint, a remarkable feature of Gaussian processes is that their behaviour is fully captured by their mean and covariance functions, which can be evaluated recursively layer by layer ([Lee et al., 2018](#)).

Interestingly, contrary to the naive belief “deeper is more expressive”, [Schoenholz et al. \(2017\)](#) pointed out that the network-equivalent Gaussian process becomes trivial as the number of layers of a network goes to infinity: the output forgets about the input, thus lacking expressive power. From an information-theoretic perspective, each randomly initialised layer acts as a noise source, and consequently, the output of the last layer loses all information from the original input.

One natural way to tackle this issue is to introduce skip connections to propagate the input information deeper into the networks. Architectures of this kind are usually referred to as residual networks (in short, ResNets). However, standard ResNets are not a viable solution in the large width and depth regime, primarily because the output tends to explode with depth. Even when normalisation factors are added to prevent divergence, the dependence on the input remains trivial. This fact implies the impossibility of effectively training deep networks unless more involved and computationally expensive normalisation techniques are used (*e.g.*, batch normalisation).

To address the above concerns, we introduce in [Hayou et al. \(2021\)](#) a new class of residual networks, named stable ResNets, which we show to preserve expressiveness and trainability even for diverging depth. The key idea is to introduce layer- and/or depth-dependent scaling factors to the ResNet blocks, allowing for better control over the noisy contribution of each layer. It is worth mentioning that the idea of a depth-dependent scaling had already been previously proposed in the literature ([De and Smith, 2020](#); [Zhang et al., 2019](#)). However, to the best of our knowledge, prior to our work no analysis of the expressiveness of the infinite-depth limit had been performed in this setting, and no layer-dependence had been considered.

This chapter provides a detailed exposition of our main theoretical findings. Section 2.3 presents basic definitions and properties of reproducing kernels and Gaussian processes. It mainly serves as a recall of results from the related literature, an exception being the discussion on kernels' expressiveness (Definition 7, Lemma 2, and Lemma 4). Section 2.4 considers the Gaussian limit of wide networks. Most of the results therein are well-known in the literature, except for Section 2.4.3, where Proposition 1 is based on one of the main results (Proposition A1) of our paper [Hayou et al. \(2021\)](#). The core of this chapter is Section 2.5, which introduces and analyses stable ResNets. Section 2.5.6 briefly introduces the neural tangent kernel and gives a quick overview of the behaviour of the stable ResNets in the NTK regime. Finally, Section 2.5.7 quickly summarises the experimental results from [Hayou et al. \(2021\)](#), validating our theory.

2.3 Mathematical preliminaries

For this whole section, unless otherwise specified, we let K be a compact set in \mathbb{R}^p (with p a positive integer) and μ a finite Borel measure on K . Moreover, x and x' will always denote arbitrary elements in K . Finally, the notation $[a : b]$ denotes the set of integers in $[a, b]$, where $a, b \in \mathbb{N}$ and $b \geq a$.

2.3.1 Kernels

The literature is rich in different (and sometimes contradicting) definitions of kernel. Usually, given a set S , one defines a kernel as a real (or complex) function Q , on $S \times S$, which is symmetric and non-negative definite. However, in the present work, we will limit ourselves to

studying continuous kernels on compact subsets of \mathbb{R}^p (often referred to as Mercer's kernels). We have hence opted (as in (Hayou et al., 2021)) for a less general definition.

Definition 1 (Kernel). *A kernel on K is a continuous function $Q : K^2 \rightarrow \mathbb{R}$ that is symmetric in its arguments (i.e., $Q(x, x') = Q(x', x)$ for all $x, x' \in K$) and non-negative definite, meaning that for all positive integers $n \geq 1$, for all subsets $\hat{K} = \{x_1 \dots x_n\} \subset K$, the Gram matrix $(Q(x_i, x_j))_{i,j}$ is non-negative definite.¹*

Kernels are closely related to integral operators.

Definition 2 (Induced integral operator). *Let Q be a kernel on K and μ a finite Borel measure on K . We define the induced integral operator $T_\mu(Q)$ on $L^2(K, \mu)$ as*

$$T_\mu(Q) \varphi(x) = \int_K T(x, x') \varphi(x') d\mu(x').$$

Lemma 1. *Let $Q : K^2 \rightarrow \mathbb{R}$ be a continuous symmetric function. Given any finite Borel measure μ on K , the induced operator $T_\mu(Q)$ is bounded, compact, and self-adjoint.² Moreover, Q is a kernel if, and only if, for all finite Borel measures μ on K , $T_\mu(Q)$ is non-negative definite, which means $\langle T_\mu(Q) \varphi, \varphi \rangle \geq 0$ for all $\varphi \in L^2(K, \mu)$.*

We refer to Section 2.6.1 for a proof of the above characterisation.

A classical result is the following decomposition (e.g., Paulsen and Raghupathi, 2016).

Theorem 3 (Mercer). *Let Q be a kernel on K and μ a fully supported finite Borel measure on K . Denote as $\{\varphi_n\}_{n \in \mathbb{N}}$ and $\{\xi_n\}_{n \in \mathbb{N}}$ the eigenfunctions and eigenvalues of $T_\mu(Q)$. The operator $T_\mu(Q)$ is trace-class (i.e., $\sum_{n \in \mathbb{N}} \xi_n < \infty$), its eigenfunctions $\varphi_n : K \rightarrow \mathbb{R}$ are all continuous, and we can write*

$$Q(x, x') = \sum_{n \in \mathbb{N}} \xi_n \varphi_n(x) \varphi_n(x'),$$

for all $x, x' \in K$, the convergence of the sum being uniform on K^2 .

Note that the above decomposition depends on the choice of the measure μ . Moreover, the result easily extends to the case of a not fully supported measure μ . Indeed, denoting with \tilde{K}

¹We recall that a $n \times n$ matrix M is non-negative definite if for every n -vector v we have $v^\top M v \geq 0$.

²A bounded self-adjoint compact operator is a linear map $T : L^2(K, \mu) \rightarrow L^2(K, \mu)$ that maps bounded sets to bounded sets whose closure is compact, and such that for every $\varphi, \psi \in L^2(K, \mu)$ we have $\langle T\varphi, \psi \rangle = \langle \varphi, T\psi \rangle$. We refer to Lang (2012) for an overview of the theoretical properties of this class of operators.

the support of μ , we have that \tilde{K} is a closed subset of a compact set, and so it is compact. Applying Theorem 3 to the restriction of Q on \tilde{K} , and extending its eigenfunctions continuously to the whole K , we see that the only modification to be made is that the convergence will be on the support of μ only.

There is an interesting link between kernels and Hilbert spaces.

Definition 3 (RKHS). *For each kernel Q on K , there exists a unique (up to isomorphisms) real Hilbert space \mathcal{H}_Q , with the following properties (Paulsen and Raghupathi, 2016):*

- *The elements of \mathcal{H}_Q are functions $K \rightarrow \mathbb{R}$.*
- *Denoting as $\langle \cdot, \cdot \rangle_Q$ the inner product of \mathcal{H}_Q , for each $x \in K$ there exists an element $k_x \in \mathcal{H}_Q$, such that $h(x) = \langle h, k_x \rangle_Q$ for all $h \in \mathcal{H}_Q$.*
- *For all $x, x' \in K$, $\langle k_x, k_{x'} \rangle_Q = Q(x, x')$.*

\mathcal{H}_Q is called the reproducing kernel Hilbert space (RKHS) of Q .

Generally, giving a concrete description of the RKHS associated with a kernel Q is not easy. However, we can easily find an explicit form for the elements k_x appearing in its definition, as we have that $k_x : x' \mapsto Q(x, x')$. In particular, we can say that \mathcal{H}_Q contains the linear span of $\{x' \mapsto Q(x, x')\}_{x \in K}$, which turns out to be a dense subset of \mathcal{H}_Q , with respect to the norm of \mathcal{H}_Q . Moreover, for any finite Borel measure μ on K , it can be shown that the RKHS of Q contains the range of $T_\mu(Q)$. We refer to Paulsen and Raghupathi (2016) for proofs and discussion of these properties.

We conclude this section by introducing the concept of feature map.

Definition 4 (Feature map). *Let Q be a kernel on K and \mathcal{H} an arbitrary real Hilbert space. A feature map for Q is a continuous map $\Phi : K \rightarrow \mathcal{H}$ such that, for all $x, x' \in K$,*

$$Q(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{H}}.$$

Note that the request of continuity in the above definition is unnecessary, as $\|\Phi(x) - \Phi(x')\|_{\mathcal{H}}^2 = Q(x, x) + Q(x', x') - 2Q(x, x')$, which vanishes whenever $x \rightarrow x'$ thanks to the continuity of the kernel.

It is straightforward that $\Phi : x \mapsto k_x$ is a feature map for Q , where $k_x : x' \mapsto Q(x, x')$. This is usually called the canonical feature map. Moreover, as a corollary of Theorem 3, we can find feature maps $K \rightarrow \ell^2$, usually referred to as Mercer's representations (Kanagawa et al., 2018): for any finite and fully supported³ μ , we consider

$$\Phi : x \mapsto \{\sqrt{\xi_n} \varphi_n(x)\}_{n \in \mathbb{N}} \in \ell^2. \quad (2.1)$$

2.3.2 Gaussian processes

We recall that a Gaussian vector is a finite collection of Gaussian random variables, such that any linear combination of its elements is still normally distributed. This concept can be extended to a family of Gaussian random variables indexed on a generic set S . In particular, we will be interested in the case where S is a compact $K \subset \mathbb{R}^p$.

Definition 5 (Gaussian process). *A Gaussian process on K is a random function $U : X \rightarrow \mathbb{R}$, such that for any finite subset $\hat{K} \subset K$, the family $\{U(x)\}_{x \in \hat{K}}$ is a Gaussian vector. A Gaussian process U is said to be centred if $\mathbb{E}[U(x)] = 0$, for all $x \in K$.*

A Gaussian process can be fully characterised by its mean and covariance functions (Dudley, 2002). In particular, we can associate to each kernel Q a centred Gaussian process.

Definition 6 (Induced Gaussian process). *Given a kernel Q on K , we define the induced Gaussian process U_Q as the centred Gaussian process on K whose covariance function is Q . More explicitly, $\mathbb{E}[U_Q(x)] = 0$ and $\mathbb{E}[U_Q(x)U_Q(x')] = Q(x, x')$ for all $x, x' \in K$.*

Mercer's theorem (Theorem 3) allows us to write the Gaussian process induced by a kernel as a converging sum, usually called Karhunen-Loève expansion.

Theorem 4 (Karhunen-Loève). *Consider a kernel Q on K and fix a fully supported finite Borel measure μ on K . Let $\{\varphi_n\}_{n \in \mathbb{N}}$ and $\{\xi_n\}_{n \in \mathbb{N}}$ be the eigenfunctions and eigenvalues of the induced operator $T_\mu(Q)$ on $L^2(K, \mu)$. Then, we can write*

$$U_Q = \sum_{n=0}^{\infty} Z_n \sqrt{\xi_n} \varphi_n, \quad (2.2)$$

³Note that for a compact set $K \subset \mathbb{R}^p$ a fully supported finite Borel measure always exists. For instance, we can find a sequence $\{x_n\}_{n \in \mathbb{N}} \subset K$ which is dense in K , and then define $\mu = \sum_{n \in \mathbb{N}} 2^{-n} \delta_{x_n}$, where δ_x is the Dirac measure with unit mass on x .

with $\{Z_n\}_{n \in \mathbb{N}}$ a family of independent standard normal random variables. The convergence is uniform on K and in squared mean: $\lim_{N \rightarrow \infty} \sup_{x \in K} \mathbb{E}[(U_Q(x) - \sum_{n=0}^N Z_n \sqrt{\mu_n} \varphi_n(x))^2] = 0$. Moreover, the sum converges in $L^2(K, \mu)$ almost surely.

A proof of the above theorem can be found in the last chapter of [Paulsen and Raghupathi \(2016\)](#), with the exception of the last statement (almost sure convergence), for which we refer to the preliminary discussion in the introduction of [Steinwart \(2019\)](#).

2.3.3 Expressiveness and universality

The main focus of this chapter is to discuss the expressiveness of neural networks in their Gaussian limit. Intuitively, the expressiveness of a Gaussian process can be interpreted as the potential of the process to express a wide range of functions. In this section, we will make this concept more rigorous.

A consequence of the almost sure L^2 -convergence in [Theorem 4](#) is the fact that, for any finite Borel measure μ on K , the samples from U_Q are in $L^2(K, \mu)$ with probability 1.⁴ We might wonder how much of this space can be explored by the process. This is the motivation for the next definition of expressiveness.

Definition 7 (Expressiveness). *Fix a finite Borel measure μ over K . A kernel Q on K is μ -expressive when, for all $\varphi \in L^2(K, \mu)$ and all $\varepsilon > 0$, its induced Gaussian process U_Q satisfies*

$$\mathbb{P}(\|U_Q - \varphi\|_2 \leq \varepsilon) > 0.$$

A kernel μ -expressive for all non-zero finite Borel measure μ on K is said to be fully expressive.

[Theorem 4](#) implies that the μ -expressiveness of Q is directly linked to the strictly positive definiteness of the operator $T_\mu(Q)$, as stated in the next lemma (see [Section 2.6.1](#) for a proof).

Lemma 2. *Let Q be a kernel and μ a non-zero finite Borel measure on K . Q is μ -expressive if, and only if, $T_\mu(Q)$ is strictly positive definite, namely $\langle T_\mu(Q) \varphi, \varphi \rangle > 0$ for all non-zero $\varphi \in L^2(K, \mu)$.*

Another way to characterise the expressiveness of a kernel is to look at the size of its RKHS. In line with the previous literature ([Micchelli et al., 2006](#); [Steinwart, 2001](#)), we call

⁴Note that this is true even if μ has not full support since the theorem applies on the support of μ , which is compact.

universal a kernel whose RKHS can approximate arbitrarily well any continuous function. To make sense of this, first notice that the continuity of Q implies that $\mathcal{H}_Q \subseteq C(K)$, the space of continuous functions on K (Paulsen and Raghupathi, 2016).

Definition 8 (Universality). *A kernel Q is said universal on K when its RKHS \mathcal{H}_Q is dense in $C(K)$, with respect to the uniform norm.*

We stated in Lemma 1 that a kernel can be characterised by the fact that it induces non-negative definite integral operators. Strictly positive definiteness corresponds to universality.

Lemma 3 (Sriperumbudur et al., 2011). *Let Q be a kernel on K . Q is a universal kernel if, and only if, for all non-zero finite Borel measures μ on K , $T_\mu(Q)$ is strictly positive definite, namely $\langle T_\mu(Q) \varphi, \varphi \rangle > 0$ for all non-zero $\varphi \in L^2(K, \mu)$.*

As a corollary of Lemma 2 and Lemma 3, universality and full expressiveness are equivalent.

Lemma 4. *A kernel Q on K is universal if, and only if, it is fully expressive.*

We conclude by a useful characterisation of universality from Micchelli et al. (2006).

Lemma 5. *Let Q be a kernel on K and $\Phi = \{\phi_n\}_{n \in \mathbb{N}} : K \rightarrow \ell^2$ a feature map, namely*

$$\sum_{n \in \mathbb{N}} \phi_n(x) \phi_n(x') = Q(x, x')$$

for all $x, x' \in K$. Assume that the convergence of the above sum is uniform on K^2 and that, for all $n \in \mathbb{N}$, $\phi_n : K \rightarrow \mathbb{R}$ is continuous. Then Q is universal if, and only if, the linear span of the family $\{\phi_n\}_{n \in \mathbb{N}}$ is dense in $C(K)$, with respect to the uniform norm.

We remark that the above lemma can be applied to the Mercer's representation (2.1) induced by a fully supported finite Borel measures on K .

2.4 Gaussian limit for neural networks

2.4.1 Simple fully-connected neural networks

Consider a simple feed-forward fully-connected neural network $F : \mathbb{R}^p \rightarrow \mathbb{R}^q$, defined via

$$\begin{aligned} F(x) &= u^L(x); \\ u^l(x) &= \frac{\sigma_w}{\sqrt{n_l}} w^l \phi(u^{l-1}(x)) + \sigma_b b^l; \\ u^0(x) &= \frac{\sigma_w}{\sqrt{p}} w^0 x + \sigma_b b^0, \end{aligned} \tag{2.3}$$

with both σ_b and σ_w strictly positive. We denote the width of u_l as n_l (so $n_L = q$) and we assume that all the parameters are randomly initialised as independent draws from a standard normal distributino $\mathcal{N}(0, 1)$.

Since all the weight matrices and the bias vectors are random objects (at least at the initialisation), every node u_i^l maps the input x to a random variable $u_i^l(x)$. It easy to check by induction that the components of $u^l(x)$ share the same distribution and are independent. Hence, for each layer $l \in [0 : L]$, we can define a random process U^l on \mathbb{R}^p , such that for every input x and index $i \in [1 : N_l]$, we have $y_i^l(x) \sim U^l(x)$.

Clearly, at the initialisation U^0 is a Gaussian process, since it is a finite linear combination of independent Gaussian processes, while for $l \geq 1$ in general U^l can be non-Gaussian. However, we can always compute recursively means and covariances. First of all, as long as $\mathbb{E}[\phi(U^{l-1}(x))] < \infty$, we are granted that $\mathbb{E}[U^l(x)] = 0$. Hence, under some weak hypothesis on the tails of ϕ , U^l is centred for all $l \in [0 : L]$. Moreover, we can get recursively the covariance functions, at least when we can ensure that $\mathbb{E}[\phi(U^{l-1}(x))\phi(U^{l-1}(x'))]$ stays bounded. Denoting as Q_l the covariance function of U^l , we have

$$\begin{aligned} Q_0(x, x') &= \mathbb{E}[U^0(x)U^0(x')] = \frac{\sigma_w^2}{p} x \cdot x' + \sigma_b^2; \\ Q_l(x, x') &= \mathbb{E}[U^l(x)U^l(x')] = \sigma_w^2 \mathbb{E}[\phi(U^{l-1}(x))\phi(U^{l-1}(x'))] + \sigma_b^2. \end{aligned} \tag{2.4}$$

2.4.2 Gaussian limit

Every node of an infinitely wide neural network behaves like a Gaussian process at the initialisation. Intuitively, the idea behind this limit is quite simple. Each node in the layer l is a sum of identically distributed random variables coming from the previous layer. When the number of nodes in the layer $l - 1$ tends to infinity, we have a sum of infinitely many terms and hence Gaussianity, thanks to the central limit theorem.⁵ The covariance of each layer can be expressed recursively via (2.4).

This chapter focuses on ReLU networks, whose activation function is the rectified linear unit $\phi : z \mapsto \max(0, z)$. In this case, there is an explicit form for $\mathbb{E}[\phi(U(x))\phi(U(x'))]$, where U is a centred Gaussian process with covariance Q (Daniely et al., 2016). To clarify this point, let us first define the correlation kernel of the process U , which is given by

$$C(x, x') = \frac{Q(x, x')}{\sqrt{Q(x, x)Q(x', x')}}. \quad (2.5)$$

For all $x, x' \in \mathbb{R}^p$, when ϕ is the rectified linear unit we have

$$\mathbb{E}[\phi(U(x))\phi(U(x'))] = \frac{Q(x, x')}{2} \left(1 + \frac{f(C(x, x'))}{C(x, x')} \right),$$

where $f : [-1, 1] \rightarrow \mathbb{R}$ is defined via

$$f : \gamma \mapsto \frac{1}{\pi} (\sqrt{1 - \gamma^2} - \gamma \arccos \gamma). \quad (2.6)$$

Hence, the equivalent Gaussian processes of an infinitely wide ReLU network have covariances given by

$$\begin{aligned} Q_0(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{p} x \cdot x'; \\ Q_l &= \sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) Q_{l-1}, \end{aligned} \quad (2.7)$$

where C_l is defined as in (2.5).

⁵As a side note, we recall that things are more delicate as we want to show the convergence of a sequence of stochastic processes; cf. the discussion in Section 1.3. However, to avoid unnecessary technicalities that are out of the scope of this thesis, in the following we will freely speak of the *equivalent Gaussian processes* of the layers of an infinitely wide network, meaning centred Gaussian processes whose covariance functions Q_l can be derived recursively, à la (2.4).

2.4.3 Expressiveness for finite depth

We now focus on the expressiveness of each layer of an infinitely wide ReLU network. We fix as input space a compact set $K \subset \mathbb{R}^p$. For all $l \in [0 : L]$, the covariance and correlation functions (Q_l and C_l) are kernels, as in Definition 1 (see Lemma 10 in Section 2.6.2).

Recalling the results and definitions of Section 2.3.3, we focus here on whether the kernels Q_l are universal, and hence fully expressive by Lemma 4. For sure, this is not the case for Q_0 . Indeed, for any finite Borel measure μ on K , $T_\mu(Q_0)$ has at most rank two. So, by Theorem 4, the realisations of its induced Gaussian process lie in a 2-dimensional space. However, it turns out that for all layers $l \geq 1$, the kernel Q_l is universal.

Proposition 1. *Fixed any compact $K \subset \mathbb{R}^p$, for $l \in [1 : L]$, Q_l is a universal kernel on K , as in Definition 8.*

The above proposition is a slight modification of Proposition 3 in our paper Hayou et al. (2021). The proof (see Section 2.6.2) can be decomposed into two parts, which are peculiar to most of the proofs of universality in this chapter. First, we show that Q_1 is universal, then that the universality of Q_{l-1} implies the universality of Q_l . The first step is the most challenging and can be seen as a corollary of the next result.

Proposition 2. *Let $K \subset \mathbb{R}^p$ be compact. Let $\tilde{f} : \gamma \mapsto \frac{\gamma}{2} + f(\gamma)$ be defined on $[-1, 1]$. Then $\tilde{f}(C_0)$, defined point-wise as $\tilde{f}(C_0)(x, x') = \tilde{f}(C_0(x, x'))$, is a universal kernel on K .*

The proof of the above proposition can be found in Section 2.6.1. The main idea is to use the Stone-Weirstrass theorem (Lang, 2012) to show the density, in the space of continuous functions, of the linear span of a feature representation of $\tilde{f}(C_0)$, and conclude by Lemma 5.

2.4.4 Infinite-depth limit

As mentioned earlier, having an excessively large number of layers in a wide neural network often results in the output being either a random constant or almost certainly discontinuous, thereby losing its connection with the input (Schoenholz et al., 2017; Hayou et al., 2019b). To explain this phenomenon mathematically, we can look at the behaviour of the functions Q_l and C_l as $l \rightarrow \infty$. All the diagonal elements of Q_l (i.e., $Q_l(x, x)$ for $x \in K$) tend to some fixed value $q_\star > 0$, independent of x . At the same time, for $x \neq x'$, $C_l(x, x')$ approaches a fixed

value $c_\star \in [-1, 1]$, which is independent of x and x' (Schoenholz et al., 2017). The fact that the final kernels lose their dependence on x and x' implies that the input information gets lost in the propagation through the network. The ordered behaviour (*i.e.*, a random constant output) comes about when $c_\star = 1$, whilst $c_\star < 1$ is at the origin of a highly discontinuous output. The values of q_\star and c_\star are determined by the parameters σ_b and σ_w . However, for a ReLU network only the ordered case is present.

Before proceeding further, it is worth clarifying how the completely inexpressive infinitely deep network can relate to its fully expressive finite-depth counterpart. Indeed, the claims of Section 2.4.3 might even be strengthened: one can show that there is an increasing hierarchy for the RKHSs of the covariance functions, in the sense that $\mathcal{H}_{Q_{l-1}} \subseteq \mathcal{H}_{Q_l}$ (cf. the proof of Proposition 1). However, there is no contradiction. To clarify this, recall Theorem 4 and its notations. The contribution of each eigenfunction φ_n in the expansion (2.2) is weighted by the square root of the eigenvalue ξ_n . When a kernel is fully expressive, all the ξ_n 's must be strictly positive. This is indeed the case for all the Q_l 's. However, a limit of strictly positive definite operators is not necessarily strictly positive and might have null eigenvalues. This is precisely what happens here, where only the first eigenvalue of the limiting kernel is non-zero. As we add more and more layers to the network, the output collapses on φ_0 (here, the constant function) since the weighted contributions from all the other eigenfunctions vanish.

All the results discussed so far regard the state of the network at its random initialisation, and one might think that the problems mentioned above become irrelevant after the training. However, these same issues make it extremely difficult to train extremely deep networks, at least with gradient-based algorithms. A very heuristic justification for this claim is that a network that is unable to see its input cannot learn anything. More rigorously, this is reflected by the fact that, as the number of layers goes to infinity, the gradient of the output (with respect to the parameters) vanishes or explodes with probability 1, depending on whether the network is in the ordered or chaotic phase (Schoenholz et al., 2017; Hayou et al., 2019b). Moreover, the neural tangent kernel (see Section 2.5.6), describing the network's evolution during the training, becomes trivial as the depth diverges (Hayou et al., 2019a).

2.4.5 Residual networks

A natural attempt to prevent the loss of input information is the introduction of skip connections. An architecture with this feature is usually called a residual network (ResNet). As a simple example, let us consider

$$\begin{aligned} u^0(x) &= \sigma_b b_0 + \frac{\sigma_w}{\sqrt{p}} w_0 x; \\ u^l(x) &= u^{l-1}(x) + \sigma_b b_l + \frac{\sigma_w}{\sqrt{N_{l-1}}} w_l \phi(u^{l-1}(x)). \end{aligned} \tag{2.8}$$

The only difference from (2.3) is the addition of the term u^{l-1} in the evaluation of u^l , which will help propagate the input information through the network. However, we will soon see that this is not enough to prevent a trivial limit as $L \rightarrow \infty$.

First, notice that the introduction of the skip connections does not affect the fact that each layer of the network becomes a Gaussian process.⁶ We get, however, a slight modification for the recurrence relation of the covariance functions, which now reads (see [Hayou et al., 2021](#))

$$\begin{aligned} Q_0(x, x') &= \frac{\sigma_w^2}{p} x \cdot x' + \sigma_b^2; \\ Q_l(x, x') &= Q_{l-1}(x, x') + \sigma_w^2 \mathbb{E}[\phi(U^{l-1}(x))\phi(U^{l-1}(x'))] + \sigma_b^2. \end{aligned}$$

As before, things can be stated more explicitly for a ReLU network:

$$\begin{aligned} Q_0(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{p} x \cdot x'; \\ Q_l &= Q_{l-1} + \sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}}\right) Q_{l-1}. \end{aligned} \tag{2.9}$$

We want to focus on the infinite-depth limit. Nevertheless, the diagonal elements of Q_l explode as $l \rightarrow \infty$. To see this, note that $C_l(x, x) = 1$ by (2.5), for all $x \in \mathbb{R}^p$. Noticing that $f(1) = 0$, we get that, for the diagonal elements of the covariance, the recursion reads $Q_l = \sigma_b^2 + (1 + \frac{\sigma_w^2}{2})Q_{l-1}$. This leads to $Q_l \geq (1 + \frac{\sigma_w^2}{2})^l Q_0$, which clearly diverges for $l \rightarrow \infty$.

We might fix this problem by considering a slightly different architecture. Fix a strictly

⁶The results in [Yang \(2019a\)](#) hold for a large class of architectures, including residual networks.

positive parameter $\lambda \in (\delta, 1)$, with $\delta = 2/(1 + \frac{\sigma_w^2}{2})$, and consider the network

$$\begin{aligned} u^0(x) &= \sigma_b B_0 + \frac{\sigma_w}{\sqrt{p}} W_0 x; \\ u^l(x) &= (1 - \lambda) u^{l-1}(x) + \lambda \left(\sigma_b B_l + \frac{\sigma_w}{\sqrt{N_{l-1}}} W_l \phi(u^{l-1}(x)) \right). \end{aligned} \quad (2.10)$$

Now, the second equation in (2.9) becomes

$$Q_l = (1 - \lambda)^2 Q_{l-1} + \lambda^2 \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) Q_{l-1} \right). \quad (2.11)$$

It is not hard to check that, in this case, the covariance Q_l does not diverge. Also, the output's gradient (with respect to the network's parameters) that explodes for the architecture (2.9) is now stabilised. Nevertheless, a problem still occurs. For both (2.9) and (2.11), C_l tends to the constant 1.⁷ As a consequence, even in this stabilised setting, an infinitely deep network at the initialisation produces a trivial output, lying in the two-dimensional space spanned by the constant function 1 and the function $x \mapsto \sqrt{Q(x, x)}$. Moreover, a constant correlation also leads to a trivial neural tangent kernel, and hence, despite the bounded gradient, the network can only fit functions belonging to some small class.

2.5 Stable ResNets

In the previous sections, we have discussed how the output of an infinitely wide neural network becomes trivial in the limit of infinite depth because too many noisy layers corrupt the input information. Introducing skip connections is insufficient to solve the problem and yields a divergent limit unless scaling factors are added. In this section, we show that it is possible to rescale the contribution of each layer in a way that solves both the expressiveness and trainability issues. We start with discussing a toy model, where we can show that the output is fully expressive in the infinite-depth limit with elementary tools from functional analysis. Later on, using the results of kernel analysis introduced in Section 2.3.3, we will establish the universality for networks acting on a generic compact $K \subset \mathbb{R}^p$. The whole analysis always focuses on ReLU networks.

⁷It is enough to see that there is a single fixed point for the diagonal terms of Q in (2.11), and hence a single fixed point ($C = 1$) for the correlation.

2.5.1 A toy model

Let $I \subset (0, +\infty)$ be a compact interval and denote as $\phi : x \mapsto \max(0, x)$ the ReLU activation function. For each $L \geq 1$, we consider the following residual architecture, where all the layers have width N ,

$$\begin{aligned} u_0(x) &= w_0 x ; \\ u_l(x) &= u_{l-1}(x) + \frac{1}{\sqrt{L}} \left(\sigma_b b_l + \frac{\sigma_w}{\sqrt{N}} w_l \phi(u_{l-1}(x)) \right), \end{aligned}$$

with $x \in I$. A part from the technical assumption that the layer 0 has no bias⁸, the difference with the architecture (2.8) is the fact that now there is a factor $1/\sqrt{L}$ that weights the contribution of each layer. Taking the limit $L \rightarrow \infty$, the introduction of this scaling brings a renormalisation which allows for a finite and expressive output.

Note that now the recursion (2.9) gets slightly modified as

$$\begin{aligned} Q_0(x, x') &= \frac{\sigma_w^2}{p} x \cdot x' ; \\ Q_l &= Q_{l-1} + \frac{1}{L} \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) \right) Q_{l-1} . \end{aligned} \tag{2.12}$$

As $L \rightarrow \infty$, the above relation can be seen as the discretised version of an ODE. Indeed, we can rescale the index l as $t(l) = l/L$, so that $t(0) = 0$ and $t(L) = 1$. It is then natural, in the limit of infinite depth, to consider t as a continuous variable, spanning the whole interval $[0, 1]$, and look at the continuous limit of (2.12):

$$\begin{aligned} \dot{q}_t(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(c_t(x, x'))}{c_t(x, x')} \right) q_t(x, x'), \\ q_0(x, x') &= \sigma_w^2 \frac{x \cdot x'}{p}, \\ c_t(x, x') &= \frac{q_t(x, x')}{\sqrt{q_t(x, x)q_t(x', x')}} . \end{aligned} \tag{2.13}$$

In Section 2.5.3, we will state results of existence and uniqueness of the solution of the above Cauchy problem, as well as uniform convergence in t and x of the discretised problem (2.12) to (2.13). Moreover, we will show that, for all $t \in [0, 1]$, both q_t and c_t are kernels in the sense

⁸This technical assumption is due to the elementary techniques involved in the proofs of the main results of this section. However, the case with bias in every layer is a particular case of the analysis in Section 2.5.3.

of Definition 1.

The existence and continuity of $t \mapsto q_t(x, x')$ in $[0, 1]$ (for all inputs x, x') is enough to claim that the output kernel q_1 keeps bounded, despite the infinite number of layers. But what about the expressiveness? It turns out that, denoted by ρ the standard Lebesgue measure on I , the following holds.

Theorem 5 (Universality). *For all $t \in (0, 1]$, the kernel q_t solving (2.13) is ρ -expressive.*

Proof's sketch. We refer to Section 2.6.3 for a full proof of the above result and only present a brief sketch here. The idea consists of first showing that there is some expressiveness for small t , then that the result can be preserved as t grows. By Lemma 2, the ρ -expressiveness of q_t is equivalent to the fact that the integral operator $T_\rho(q_t)$ on $L^2(I, \rho)$ is strictly positive definite. Hence we need to show that, for all non-zero $\varphi \in L^2(I, \rho)$, we have $\langle q_t \varphi, \varphi \rangle > 0$ if $t > 0$. First, we can look at what happens when $t \simeq 0^+$, namely when t is greater than 0 but still very small. Using the fact that $c_0 = 1$ (at least with the current definition of q_0) and $f(1) = 0$, we can expand q_t around $t = 0$ as

$$q_t(x, x') = e^{\sigma_w^2 t/2} x x' + \frac{2\sigma_w^2}{\sigma_b^2} (e^{\sigma_w^2 t/2} - 1) + \frac{\sigma_b^3 \sigma_w^2}{15\pi} \frac{|x - x'|^3}{(x x')^2} t^{5/2} + o(t^{5/2}).$$

A direct study the integral operator induced by $(x, x') \mapsto \frac{|x-x'|^3}{(x x')^2}$ leads to the next result.

Proposition 3. *For any non-zero $\varphi \in L^2(I, \rho)$, there is a $t_\varphi \in (0, 1]$ such that $\langle T_\rho(q_t) \varphi, \varphi \rangle > 0$, for all $t \in (0, t_\varphi)$.*

Once established that there is some sort of strictly positive definiteness near $t = 0$, we need to show that this is preserved as t grows. However, this is not hard since it can be proven that \dot{q}_t is a kernel, so $T_\rho(\dot{q}_t)$ is non-negative definite. We can then show that $\frac{d}{dt} \langle T_\rho(q_t) \varphi, \varphi \rangle = \langle T_\rho(\dot{q}_t) \varphi, \varphi \rangle$, and hence the next proposition follows.

Proposition 4. *For all $\varphi \in L^2(I, \rho)$, the map $t \mapsto \langle T_\rho(q_t) \varphi, \varphi \rangle$ is non decreasing on $[0, 1]$.*

Propositions 3 and 4 are enough to prove the positive definiteness of q_t . □

To give a more concrete idea of the behaviour of our toy model, we report empirical results from a network with depth 200 and width 200, mapping inputs in $[0.1, 0.5]$ to real values.⁹

⁹Here, $n_L = 200$ and the output of the network is $y = u^L \cdot v$, where v is a random vector whose elements are independent and identically distributed as $\mathcal{N}(0, 1/L)$, so that the covariance function of the output corresponds

Looking at the eigendecomposition of the final kernel Q_L , obtained numerically from (2.12), we see in Figure 2.1 that the eigenvalues ξ_n decays polynomially, without vanishing. Moreover, it is interesting that the eigenfunctions are reminiscent of the Fourier basis. We can compare the analytical results with the actual behaviour of the network. The output can be decomposed on the eigenbasis of Q_L . From Theorem 4, we expect the coefficients of this decomposition to be independent realisations of centred Gaussian random variables whose variances are the eigenvalues of Q_L . Hence, renormalising each coefficient by the square root of the respective eigenvalue, we should obtain a list of independent draws from a standard normal distribution. This is shown graphically by the histogram in Figure 2.2 (blue bins).

2.5.2 Layer- and depth-dependent coefficients

The previous toy example shows that getting a finite and expressive limit as the depth grows is possible. The main idea is to rescale the various layers' contributions to control the initialisation's noise. We achieved this by introducing a scaling factor $1/\sqrt{L}$ for all layers. However, we can now consider a more general setting, where the scaling factors can depend both on the depth of the network and on the layer index. We hence introduce the following

to the one of the last layer, namely Q_L .

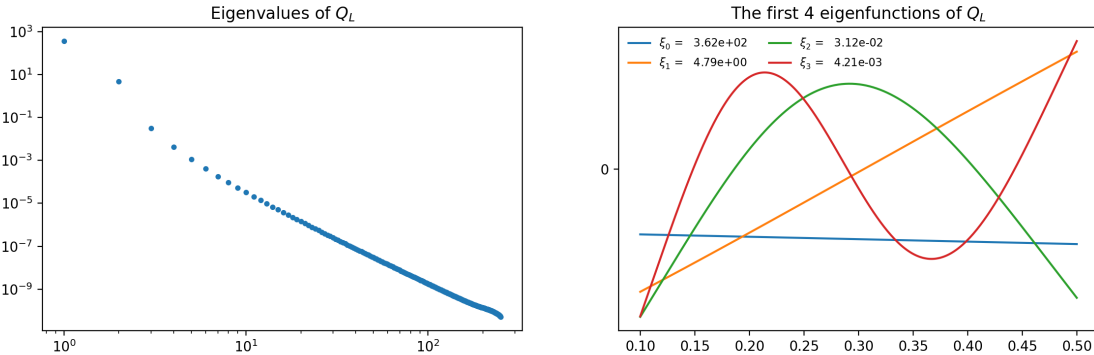


Figure 2.1: Eigendecomposition of the output kernel. The input space $I = [0.1, 0.5]$ has been discretised in 251 points, and the eigenvalues and eigenvectors of the Gram matrix of Q_L on this discretisation have been evaluated numerically. On the left are the eigenvalues plotted against their rankings. Notice the asymptotic polynomial decay. On the right are the principal four eigenvectors, showing a Fourier-like behaviour.

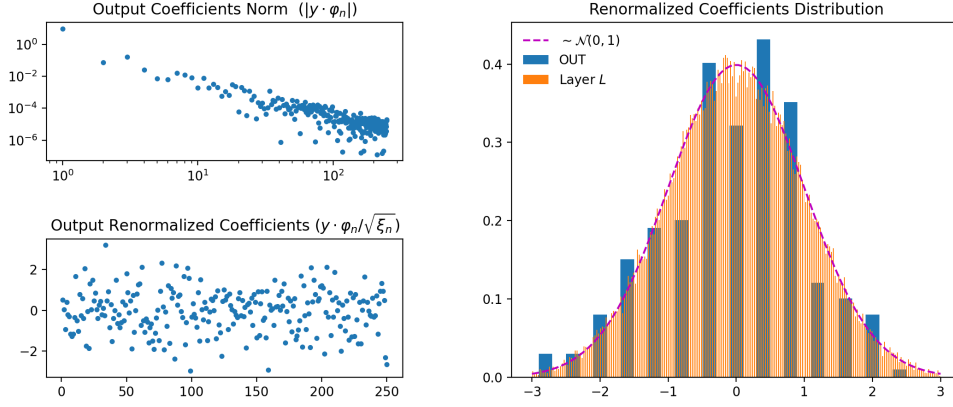


Figure 2.2: On the top left the absolute value of the coefficients $y \cdot \varphi_n$, where y is the network’s output, is plotted against $n + 1$, showing an average decrease proportional to $\sqrt{\xi_n}$. The same coefficients are renormalised on the bottom left, looking like independent draws from $\mathcal{N}(0, 1)$. This fact is confirmed in the histogram on the right, where the blue bins represent the distribution of the normalised coefficients. Noticing that the nodes of the outer layer L are independent random processes with the same law as the output, we expect the renormalised coefficients of all the outer-layer nodes to be independent draws from $\mathcal{N}(0, 1)$. This is confirmed by their empirical distribution (orange bins).

ResNet architecture:

$$\begin{aligned}
 u^0(x) &= \sigma_b b_0 + \frac{\sigma_w}{\sqrt{p}} w_0 x; \\
 u^l(x) &= u^{l-1}(x) + \lambda_{l,L} \left(\sigma_b b_l + \frac{\sigma_w}{\sqrt{N_{l-1}}} w_l \phi(u^{l-1}(x)) \right),
 \end{aligned} \tag{2.14}$$

for some depth $L \geq 1$ and non negative scaling coefficients $\{\lambda_{l,L}\}_{l \in [1:L]}$.

We are interested in the Gaussian infinite-width limit and, fixed L , we can find the covariance functions of the network recursively:

$$\begin{aligned}
 Q_0(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{p} x \cdot x'; \\
 Q_l &= Q_{l-1} + \lambda_{l,L}^2 \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) Q_{l-1} \right).
 \end{aligned} \tag{2.15}$$

Recall that, for a standard ResNet architecture, the diagonal elements of the covariance function explode exponentially with the depth: we had found in Section 2.4.5 that $Q_l(x, x) \geq (1 + \frac{\sigma_w^2}{2})^l Q_0$. This behaviour also entails that the network’s gradient (with respect to the parameters) explodes with the depth (Zhang et al., 2017). The following Proposition shows that introducing the scaling factors can stabilise the network’s output. We proved a similar result for the network’s gradient in Hayou et al. (2021).

Proposition 5 (Output’s stability). *Let $K \subset \mathbb{R}^p$ be a compact set. There exists a constant*

$\Gamma > 0$ such that, for all integers $L \geq 1$, for any non-negative coefficients $\{\lambda_{l,L}\}_{l \in [1:L]}$, the kernels Q_l , given by (2.15), satisfy

$$\sup_{l \in [0:L]} \sup_{(x,x') \in K^2} |Q_l(x,x')| \leq \left(\Gamma + \sigma_b^2 \sum_{l=1}^L \lambda_{l,L}^2 \right) e^{\frac{\sigma_w^2}{2} \sum_{l=1}^L \lambda_{l,L}^2}.$$

Proof. Fix the depth $L \geq 1$ and the scaling factors $\{\lambda_{l,L}\}_{l \in [1:L]}$. It is enough to show our claim for the diagonal terms since $|Q_l(x,x')| \leq \sqrt{Q_l(x,x)Q_l(x',x')}$ by the Cauchy-Schwarz inequality. Fix $x \in K$ and let $\Gamma = \sup_{x' \in K} Q_0(x',x')$. Define $\alpha_l = \frac{\sigma_w^2}{2} \lambda_{l,L}^2$ and $\beta_l = \sigma_b^2 \lambda_{l,L}^2$, so that $Q_l(x,x) = (1 + \alpha_l) Q_{l-1}(x,x) + \beta_l$. By induction, one can show that for all $l \in [1:L]$

$$Q_l(x,x) = Q_0(x,x) \prod_{k=1}^l (1 + \alpha_k) + \sum_{k=1}^l \beta_k \prod_{j=1}^k (1 + \alpha_j) \leq \left(Q_0(x,x) + \sum_{k=1}^L \beta_k \right) \prod_{k=1}^L (1 + \alpha_k).$$

Hence we conclude using the fact that $\prod_{k=1}^L (1 + \alpha_k) \leq \exp \sum_{k=1}^L \alpha_k$. \square

Now, let us consider a sequence of infinitely wide networks with increasing depths: for each depth $L \geq 1$, we fix the scaling factors $\{\lambda_{l,L}\}_{l \in [1:L]}$ and we consider the infinite-width limit of (2.14). We say that this is a *stable sequence* when

$$\sup_{L \geq 1} \sum_{l=1}^L \lambda_{l,L}^2 < \infty. \quad (2.16)$$

As a corollary of Proposition 5, this condition means that there is a uniform bound for all the kernels of all the networks in the sequence. In particular, if we can define an infinite-depth limit for this network sequence, we can expect all its kernels to be bounded.

However, in general, it can be hard to define a “limit” for a sequence of networks, no matter whether or not it is a stable sequence. We will restrict our analysis to two kinds of scalings, allowing us to easily make sense of the limit of infinite depth. The first case is the *uniform scaling*, given by $\lambda_{l,L} = 1/\sqrt{L}$, for all $L \geq 1$ and $l \in [1:L]$. Since $\sum_{l,L} \lambda_{l,L}^2 = 1$ for every L , (2.16) holds and hence the sequence is stable. Note that this setting is as in Section 2.5.1. The other possibility that we consider is what we call a *sequential scaling*: we fix a non-negative sequence $\{\lambda_l\}_{l \in \mathbb{N}}$, and for all $L \geq 1$ and $l \in [1:L]$ we set $\lambda_{l,L} = \lambda_l$. The corresponding sequence of networks will be stable if, and only if, $\sum_{l=1}^{\infty} \lambda_l^2 < \infty$.

In the next two sections, we will discuss the uniform and sequential scalings, showing that

the infinite-depth limit is well-defined (at least in terms of covariance functions) and fully expressive.

2.5.3 Uniform scaling

We consider a sequence of infinitely wide ResNets with increasing depth, whose architectures are given by (2.14) with scaling coefficients $\lambda_{l,L} = 1/\sqrt{L}$, for all $l \in [1 : L]$. As we are considering the Gaussian limit, from (2.15) we know that the kernels follow

$$\begin{aligned} Q_0(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{p} x \cdot x'; \\ Q_l &= Q_{l-1} + \frac{1}{L} \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) Q_{l-1} \right). \end{aligned} \quad (2.17)$$

As we discussed for the toy model of Section 2.5.1, the layer index l can be rescaled as $l \mapsto t(l) = l/L$. When $L \rightarrow \infty$, it is natural to consider t as a continuous variable spanning the interval $[0, 1]$. With this in mind, it makes sense to look at the continuous version of (2.17):

$$\begin{aligned} \dot{q}_t(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(c_t(x, x'))}{c_t(x, x')} \right) q_t(x, x'), \\ q_0(x, x') &= \sigma_b^2 + \sigma_w^2 \frac{x \cdot x'}{p}, \\ c_t(x, x') &= \frac{q_t(x, x')}{\sqrt{q_t(x, x)q_t(x', x')}}. \end{aligned} \quad (2.18)$$

As stated in the following results, for any inputs x, x' in a compact domain K , the solution of the above Cauchy problem exists and is unique. Moreover, it is regular enough to ensure that q_t and c_t are kernels, in the sense of Definition 1, for all t .

Lemma 6 (Existence and uniqueness). *For any x, x' in K , the solution of (2.18) is unique and well defined for all $t \in [0, 1]$. The maps $(x, x') \mapsto q_t(x, x')$ and $(x, x') \mapsto c_t(x, x')$ are Lipschitz continuous on K^2 and c_t takes values in $[-1, 1]$. Moreover, both q_t and c_t are kernels, in the sense of Definition 1.*

Clearly, for finite L , (2.18) is an approximation of (2.17). However, we have the following.

Lemma 7 (Convergence to the continuous limit). *Let $Q_{l|L}$ be the covariance function of the*

layer l in a net of $L + 1$ layers $[0 : L]$, and q_t be the solution of (2.18), then

$$\lim_{L \rightarrow \infty} \sup_{l \in [0:L]} \sup_{(x,x') \in K^2} |Q_{l|L}(x, x') - q_{t=l/L}(x, x')| = 0.$$

The two statements above are proved in Section 2.6.4. Note that equivalent claims can be stated (with analogous proofs) for the slightly different setting of Section 2.5.1.

The next result shows that the kernel q_t is universal for $t > 0$. In particular, by Lemma 4, these kernels are fully expressive, in the sense of Definition 7.

Theorem 6 (Universality). *Let $K \subset \mathbb{R}^p$ be compact. For any $t \in (0, 1]$, the solution q_t of (2.18) is a universal kernel on K .*

Theorem 6 is proved in Section 2.6.5. The main idea is to show that the integral operator $T_\mu(q_t)$ is strictly positive definite for all finite Borel measure μ on K , and then use Lemma 3. For the proof of the positive definiteness of the integral operator, the main approach is similar to the one of Theorem 5's proof: first, one looks at the case $t \simeq 0^+$; then, shows that the property is not lost for larger t . The major difference, always referring to the proof of Theorem 5, is in the technique used for studying the case $t \simeq 0^+$, which is based on the fact that q_0 is universal, as a consequence of Proposition 2.

2.5.4 Sequential scaling

We fix a non-negative sequence $\{\lambda_l\}_{l \in \mathbb{N}}$, such that $\sum_{l \geq 1} \lambda_l^2 < \infty$, and let $\lambda_{l,L} = \lambda_l$ for all $L \geq 1$ and $l \in [1 : L]$. The recursion (2.15) now reads

$$\begin{aligned} Q_0(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{p} x \cdot x'; \\ Q_l &= Q_{l-1} + \lambda_l^2 \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) Q_{l-1} \right). \end{aligned} \tag{2.19}$$

Note that the discussion of Section 2.4.3 applies, so we have an equivalent of Proposition 1, stating that, for all $l \in \mathbb{N}$, the kernel Q_l is universal. However, the stability of our sequence implies that, as $L \rightarrow \infty$, we get the convergence to a universal kernel.

Proposition 6 (Convergence and universality). *Fix a compact K on \mathbb{R}^p , consider the sequence of kernels on K defined in (2.19), with $\sum_{l \geq 1} \lambda_l^2 < \infty$. The sequence converges uniformly on K to a universal kernel Q_∞ .*

The proof does not present major technical difficulties and is reported in Section 2.6.6. We remark that it leverages that $Q_l - Q_{l-1}$ is a kernel, which was not true for the stabilised ResNet (2.11).

2.5.5 Expressiveness with no bias

So far, we have been considering only networks with biases. For ReLU architectures, this is a fundamental assumption in order to achieve expressiveness on a generic compact. Indeed, the output of a ReLU network with no bias is a positive homogeneous function of its input, which means that $F(\alpha x) = \alpha F(x)$ for all $\alpha \geq 0$. However, when restricting to the case $K = \mathbb{S}^{d-1}$, the unit sphere of \mathbb{R}^d (for $d \geq 2$), it is possible to obtain the same universality results for the kernels, even when no bias is present.

Proposition 7. *Let q_t be the solution of the Cauchy problem (2.18) with $\sigma_b = 0$. Then, for all $t > 0$, q_t is universal on \mathbb{S}^{d-1} . Let Q_l be the solution of (2.19) with $\sigma_b = 0$. The sequence $\{Q_l\}_{l \geq 0}$ converges uniformly to a universal kernel Q_∞ on \mathbb{S}^{d-1} , and for all $l \geq 2$, the kernel Q_l is universal on \mathbb{S}^{d-1} .*

Contrary to most of our previous results of universality, we cannot rely on Proposition 2 to show the above claims. Indeed, our proof (see Section 2.6.7) relies on the expansion in spherical harmonics, which, as suggested by its name, is peculiar to the sphere.

2.5.6 Neural tangent kernel

Most of the results discussed so far are limited to a network at its initialisation. However, Jacot et al. (2018) showed that a particular choice of parametrisation for an infinitely wide neural network leads to simple dynamics under a continuous time gradient descent training. Indeed, it is possible to study the time evolution of the network's output, which is governed by a kernel gradient descent via the neural tangent kernel (NTK). The expressiveness of this kernel is linked to the one of the network at the initialisation, and it is crucial to determine which functions can be learnt.

The NTK of a neural network is defined as

$$\hat{\Theta}_h^{ij}(x, x') = \nabla F_h^i(x) \cdot \nabla F_h^j(x'), \quad (2.20)$$

where F_h^i and F_h^j are the i -th and j -th components of the network’s output, and ∇ denotes the gradient with respect to the network’s parameters h . When trained via gradient descent to optimise the empirical loss \mathcal{L}_s (i.e., $\partial_t h_t = -\nabla \mathcal{L}_s(h_t)$)¹⁰, the network follows the dynamics

$$\partial_t F_{h_t}^i(x) = -\frac{1}{m} \sum_{(x', y') \in s} \sum_j \hat{\Theta}_{h_t}^{ij}(x, x') \partial_{F_{h_t}^j} \ell(h, y'),$$

where m is the dimension of the training data set s .

Jacot et al. (2018) showed that the NTK of a suitably randomly initialised feed-forward architecture tends to a deterministic kernel, independent of h and constant during the training. The limiting NTK is diagonal in the indexes i and j , and all its diagonal elements are equal. In particular, we can represent it with a single scalar function Θ , which is symmetric in its two arguments and, if continuous, is a kernel in the sense of Definition 1.

For completeness, we mention that the previous results hold under some additional technical assumptions. In particular, a relevant role is played by the gradient independence assumption (GIA), which requires the parameters used for the forward propagation to be independent of those used in the backward propagation used to evaluate the gradient. However, for a broad range of architectures, Yang (2020a,b) showed that this hypothesis holds if the parameters of the network’s last layer are not shared with the other layers (simple GIA check). This is the case for all the networks that we consider here.

The NTK characterises the class of functions that an infinitely wide network can learn. To see this, consider the simple setting of a network with a real output that is trained with a quadratic loss function: $\ell(h, z) = \frac{1}{2}(F_h(x) - y)^2$, where z denotes the instance-label pair (x, y) . Let us write as $\Theta(X, X)$ the Gram matrix of Θ on s , namely the matrix $(\Theta(x, x'))_{z, z' \in s}$. The network can fit any dataset s such that $\Theta(X, X)$ is non-singular. Indeed, denoting as $g(X)$ the vector $\{g(x)\}_{z \in s}$ (for a generic mapping g) and as Y the vector of labels $\{y\}_{z \in s}$, we have (Jacot et al., 2018)

$$F_{h_t}(x) - F_{h_0}(x) = \Theta(x, X) \Theta(X, X)^{-1} (\text{Id} - e^{-\Theta(X, X)t})(Y - F_{h_0}(X)),$$

which implies that $F_{h_t}(X)$ converges to Y as $t \rightarrow \infty$. As a corollary of Lemma 3, whenever

¹⁰Here t represents the “training time”. h_0 is the initial value of the parameters, which evolves as h_t under the training.

Θ is a universal kernel (and s is a set of distinct elements), the matrix $\Theta(X, X)$ is strictly positive definite (and hence non-singular), namely the network can learn any finite dataset.

As shown by [Jacot et al. \(2018\)](#), for an infinitely wide feed-forward network it is possible to evaluate the NTK Θ recursively, by suitably defining a kernel Θ_l for each layer l . By slightly adapting this approach, in [Hayou et al. \(2021\)](#) we showed that the NTK of a stable ReLU ResNet is given by $\Theta = \Theta_L$, where

$$\begin{aligned} \Theta_0 &= Q_0; \\ \Theta_{l+1} &= \Theta_l + \lambda_{l+1,L}^2 \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_l)}{C_l} \right) Q_l + \frac{\sigma_w^2}{2} (1 + f'(C_l)) \Theta_l \right). \end{aligned} \quad (2.21)$$

With arguments equivalent to those in the proof of Lemma 10, we showed in [Hayou et al. \(2021\)](#) that, for all l , Θ_l is a kernel, in the sense of Definition 1.

In the infinitely deep limit, the same problems affecting the covariance functions also occur for the NTK: the limit becomes trivial for both residual and feed-forward architectures, and it explodes in the unnormalised residual case ([Hayou et al., 2019a](#)). This is not the case for the stable ResNets, which achieve a finite and fully expressive limit in both the uniform and sequential case. We state this in the next proposition, which we proved in [Hayou et al. \(2021\)](#) (see Propositions 8 and 9 therein) using the same techniques described in this thesis to prove the results of universality.

Proposition 8 (Universal NTK; [Hayou et al., 2021](#)). *Let $\sigma_b > 0$ and $K \subset \mathbb{R}^p$ be an arbitrary compact, or $\sigma_b = 0$ and $K = \mathbb{S}^{d-1}$. For the sequential scaling, Θ_L converges uniformly over K^2 to a universal kernel Θ_∞ on K . For the uniform scaling, the NTK recursion (2.21) admits the continuous formulation*

$$\begin{aligned} \dot{\theta}_t(x, x') &= \dot{q}_t(x, x') + \frac{\sigma_w^2}{2} (1 + f'(c_t(x, x'))) \theta_t(x, x'); \\ \theta_0 &= q_0, \end{aligned}$$

where $f' : \gamma \mapsto -\frac{1}{\pi} \arccos \gamma$. For every $t > 0$, θ_t is a universal kernel on K .

2.5.7 A few comments on the empirical results

For the sake of conciseness, in this chapter we have focused on the theoretical results of expressiveness that we established in [Hayou et al. \(2021\)](#). However, the paper also discusses a few experimental results, highlighting the performance improvement brought about by uniform and sequential scalings. The stable ResNets are compared with standard residual architectures on three standard image-recognition tasks: CIFAR-10, CIFAR-100, and TinyImagenet. We tested convolutional ResNets of different depths (32, 50, and 104) and found that the stable ResNets consistently outperform their standard counterparts, with a performance gap which tends to increase with the depth.

Moreover, we gave further experimental support to our theoretical findings in the context of Gaussian process kernel regression, which can be applied directly to the kernels of the infinite-width limit. We have compared the output covariance function of scaled and unscaled ResNets architectures. The results on MNIST and CIFAR-10 show that the performance of the kernels generated by our scaled architectures keeps almost unchanged while the depth varies from 50 to 1000 layers. On the hand, the results coming from the kernels obtained via standard ResNets suffer from degradation in test accuracy as the depth grows.

More details on the experimental settings and results can be found in Section 7 and Appendix A7 of [Hayou et al. \(2021\)](#).

2.6 Omitted proofs

2.6.1 Kernels

Lemma 1. *Let $Q : K^2 \rightarrow \mathbb{R}$ be a continuous symmetric function. Given any finite Borel measure μ on K , the induced operator $T_\mu(Q)$ is bounded, compact, and self-adjoint. Moreover, Q is a kernel if, and only if, for all finite Borel measures μ on K , $T_\mu(Q)$ is non-negative definite, which means $\langle T_\mu(Q) \varphi, \varphi \rangle \geq 0$ for all $\varphi \in L^2(K, \mu)$.*

Proof. Let $Q : K^2 \rightarrow \mathbb{R}$ be continuous and symmetric. Then, $T_\mu(Q)$ is a bounded compact self-adjoint operator ([Lang, 2012](#)). Assume that Q is a kernel and fix a finite Borel measure μ on K . Let \tilde{K} denote the support of μ , which is a compact set since it is closed and in K . By

Theorem 3, we can find continuous real functions $\{F_k\}_{k \in \mathbb{N}}$ on K , such that for all $x, x' \in \tilde{K}$

$$Q(x, x') = \sum_{k=0}^{\infty} F_k(x) F_k(x')$$

and the convergence is uniform on \tilde{K}^2 . The continuity of the U^k 's implies that they can be seen as elements of $L^2(K, \mu)$. Moreover, the uniform convergence on the support of μ , along with the fact that $\mu(K) < \infty$, implies the convergence of the sum with respect to the $L^2(K, \mu)$ operator norm. In particular, $T_\mu(K)$ is a limit of non-negative definite operators, and hence it is non-negative definite.

Now, assume that, for all finite Borel measure μ , $T_\mu(Q)$ is non-negative definite. Chosen a finite set $\{x_1 \dots x_n\} \subset K$, in particular we have that $\mu = \sum_{i=1}^n \delta_{x_i}$ is a finite Borel measure (where δ_x is the Dirac measure with unit mass on $x \in K$). Hence $T_\mu(Q)$ is equivalent to the matrix $(Q(x_i, x_j))_{i,j}$. We conclude that Q is a kernel. \square

Lemma 2. *Let Q be a kernel and μ a non-zero finite Borel measure on K . Q is μ -expressive if, and only if, $T_\mu(Q)$ is strictly positive definite, namely $\langle T_\mu(Q) \varphi, \varphi \rangle > 0$ for all non-zero $\varphi \in L^2(K, \mu)$.*

Proof. Without loss of generality, we can suppose that μ is fully supported on K since, if this is not the case, it is enough to consider the restriction to the support of μ , which is still a compact set. Denote as $\{\xi_n\}_{n \in \mathbb{N}}$ and $\{\varphi_n\}_{n \in \mathbb{N}}$ the eigenvalues and the orthonormal eigenbasis of $T_\mu(Q)$. For any $\varphi \in L^2(K, \mu)$, by Theorem 4 we have that $\|\sum_{n=0}^N Z_n \sqrt{\xi_n} \varphi_n - \varphi\|_2^2$ converges in squared mean to $\|U_Q - \varphi\|_2^2$, for $N \rightarrow \infty$, where $\{Z_n\}_{n \in \mathbb{N}}$ are iid standard normal random variables. Now, let $\varphi = \sum_{n=0}^N a_n \varphi_n$ for some integer $N \geq 1$ and some real coefficients $\{a_0 \dots a_N\}$. We have (with convergence in squared mean)

$$\|U_Q - \varphi\|_2^2 = \sum_{n=0}^N \left(Z_n \sqrt{\xi_n} - a_n \right)^2 + \sum_{n=N+1}^{\infty} \xi_n Z_n^2.$$

For $n \in [0 : N]$, we can define the interval $I_n = \left[\frac{a_n}{\sqrt{\xi_n}} - \frac{\varepsilon}{\sqrt{2(N+1)\xi_n}}, \frac{a_n}{\sqrt{\xi_n}} + \frac{\varepsilon}{\sqrt{2(N+1)\xi_n}} \right]$, so that, for all $z \in I_n$ we have $(z\sqrt{\xi_n} - a_n)^2 \leq \frac{\varepsilon^2}{2(N+1)}$. Since all these intervals are non-empty, we get

$$\mathbb{P} \left(\sum_{n=0}^N \left(Z_n \sqrt{\xi_n} - a_n \right)^2 \leq \frac{\varepsilon^2}{2} \right) \geq \prod_{n=0}^N \mathbb{P}(Z_n \in I_n) > 0.$$

On the other hand, we have that

$$\delta_N = \mathbb{E} \left[\sum_{n=N+1}^{\infty} \xi_n Z_n^2 \right] = \sum_{n=N+1}^{\infty} \xi_n.$$

By Theorem 3, $T(Q)$ is trace class and hence δ_N vanishes as $N \rightarrow \infty$. By Markov's inequality

$$\mathbb{P} \left(\sum_{n=N+1}^{\infty} \xi_n Z_n^2 \geq \frac{\varepsilon^2}{2} \right) \leq \frac{2\delta_N}{\varepsilon^2}$$

and we can conclude that $\mathbb{P}(\|U_Q - \varphi\|_2 \leq \varepsilon) > 0$ for N large enough.

For a general $\varphi = \sum_{n=0}^{\infty} a_n \varphi_n$, let $\varphi_N = \sum_{n=0}^N a_n \varphi_n$. Since $\{\varphi_n\}_{n \in \mathbb{N}}$ is a basis of $L^2(K, \mu)$, fixed $\varepsilon > 0$, it is possible to find an N such that $\|\varphi - \varphi_N\|_2 \leq \varepsilon/2$ and $\mathbb{P}(\|\varphi_N - U_Q\|_2 \leq \varepsilon/2) > 0$, and so we conclude that Q is μ -expressive.

To show the other implication, assume that there is $m \in \mathbb{N}$ such that $\xi_m = 0$. Let V be the closure in $L^2(K, \mu)$ of the linear span generated by the eigenfunctions $\{\varphi_n\}_{n \neq m}$. By Theorem 4, we know that $\mathbb{P}(U_Q \in V) = 1$, and so $\mathbb{P}(\|U_Q - \varphi_m\|_2 < 1) = 0$ and we conclude. \square

Lemma 8. *Let C be a kernel on K , such that $|C(z)| \leq 1$ for all $z \in K$. Consider a non-negative real sequence $\{\alpha_n\}_{n \in \mathbb{N}}$, and assume that*

$$g(\gamma) = \sum_{k=0}^{\infty} \alpha_k \gamma^k$$

converges uniformly on $[-1, 1]$. Then, for all finite Borel measure μ on K , $T_\mu(g(C))$ is a non-negative definite compact operator, and in particular, $g(C)$ is a kernel.

Proof. Fix a finite Borel measure μ on K and notice that $g(C)$ is continuous and symmetric (as a uniform limit of continuous and symmetric functions). Moreover, since the Taylor expansion of g around 0 converges uniformly on $[-1, 1]$, and since $|C(z)| \leq 1$ for all $z \in K$, we have that $T_\mu(g(C)) = \sum_{k \in \mathbb{N}} \alpha_k T_\mu(C^k)$, the sum converging with respect to the operator norm on $L^2(K, \mu)$. Due to the Schur product theorem, the product of two kernels is still a kernel. Consequently, it is easy to prove by induction that $T_\mu(C^k)$ is non-negative definite for all k . Hence $T_\mu(g(C))$ is the converging limit of a sum of compact non-negative definite operators. We conclude by Lemma 1. \square

Lemma 9. *The function $f : [-1, 1] \rightarrow \mathbb{R}$ is an analytic function on $(-1, 1)$, whose expansion $f(\gamma) = \sum_{n \in \mathbb{N}} \alpha_n \gamma^n$ converges uniformly on $[-1, 1]$. Moreover, $\alpha_n > 0$ for all even $n \in \mathbb{N}$, $\alpha_1 = -1/2$ and $\alpha_n = 0$ for all odd $n \geq 3$.*

Proof. All claims are not hard to prove and are well known (e.g., [Daniely et al., 2016](#)). \square

Proposition 2. *Let $K \subset \mathbb{R}^p$ be compact. Let $\tilde{f} : \gamma \mapsto \frac{\gamma}{2} + f(\gamma)$ be defined on $[-1, 1]$. Then $\tilde{f}(C_0)$, defined point-wise as $\tilde{f}(C_0)(x, x') = \tilde{f}(C_0(x, x'))$, is a universal kernel on K .*

Proof. First notice that $c_0(x, x') = \frac{1 + \zeta x \cdot x'}{\sqrt{(1 + \zeta \|x\|^2)(1 + \zeta \|x'\|^2)}}$, where $\zeta = \sigma_w^2 / \sigma_b^2$. For $n \in \mathbb{N}$, define $p_n : (x, x') \mapsto c_0(x, x')^{2n}$, with the convention that $p_0 \equiv 1$. It is easy to verify that c_0 is a kernel. Consequently, p_n is a kernel for all n since it is a product of kernels. From [Lemma 9](#), we can write

$$\tilde{f}(c_0) = \sum_{n \in \mathbb{N}} \alpha_n p_n,$$

the sum converging uniformly on K^2 , with $\alpha_n > 0$ for all $n \in \mathbb{N}$. By [Lemma 8](#), $\tilde{f}(c_0)$ is a kernel. Now, for each n , we have

$$p_n(x, x') = \frac{1}{(1 + \zeta \|x\|^2)^n (1 + \zeta \|x'\|^2)^n} \sum_{k=0}^{2n} \omega_{k,n} (x \cdot x')^k,$$

where the coefficients $\omega_{k,n}$ are all strictly positive, explicitly $\omega_{k,n} = \zeta^k \binom{n}{k}$. Expanding the inner product $x \cdot x'$, we can express p_n in the form

$$p_n(x, x') = \sum_{J \in \mathcal{J}_n} \beta_{J,n} A_{J,n}(x) A_{J,n}(x'),$$

where $\mathcal{J}_n = \{(j_1 \dots j_d) \in \mathbb{N}^d : \sum_{i=1}^d j_i \in [0 : 2n]\}$, all the coefficients $\beta_{J,n}$ are strictly positive and the $A_{J,n}$ are defined as

$$A_{J,n}(x) = \frac{x_1^{j_1} \dots x_d^{j_d}}{(1 + \zeta \|x\|^2)^n}.$$

Hence we can write $\tilde{f}(c_0)$ as

$$\tilde{f}(c_0)(x, x') = \sum_{n \in \mathbb{N}} \sum_{J \in \mathcal{J}_n} \alpha_n \beta_{J,n} A_{J,n}(x) A_{J,n}(x').$$

For any $n, n' \in \mathbb{N}$, $J \in \mathcal{J}_n$, $J' \in \mathcal{J}_{n'}$, it is clear that $A_{J,n}A_{J',n'} = A_{J'',n+n'}$, where J'' is some element in $\mathcal{J}_{n+n'}$. As a consequence, the linear span of the family $\{A_{J,n}\}_{n \in \mathbb{N}, J \in \mathcal{J}_n}$ is an algebra \mathcal{A} (which is a subalgebra of $C(K)$ since all the $A_{J,n}$ are continuous). Moreover, $A_{(0\dots 0),0} \equiv 1$, so that \mathcal{A} contains a constant, and it is straightforward to check that \mathcal{A} separates points, which means that for all distinct $x, x' \in K$ there exists $a \in \mathcal{A}$ such that $a(x) \neq a(x')$. Then, from the Stone-Weierstrass theorem (Lang, 2012), \mathcal{A} is dense in $C(K)$ with respect to the uniform norm.

For all $n \in \mathbb{N}$, all $J \in \mathcal{J}_n$, let $\theta_{J,n} = \sqrt{\alpha_n \beta_{J,n}}$. Define a bijection $\iota : \mathbb{N} \rightarrow \{(n, J) : n \in \mathbb{N}, J \in \mathcal{J}_n\}$ and let $\Phi_n = \theta_{\iota(n)} A_{\iota(n)}$. For all $x \in K$, we have that $\Phi(x) = \{\Phi_n(x)\}_{n \in \mathbb{N}} \in \ell^2$, since $p_n(x, x) < \infty$. We conclude that Φ is a feature map for $\tilde{f}(c_0)$, and the density of the linear span of $\{\Phi_n\}_{n \in \mathbb{N}}$ allows us to claim that the kernel is universal on K (cf. Theorem 7 in Micchelli et al., 2006). \square

2.6.2 Finite depth

Lemma 10. *Define by recursions the functions Q_l and C_l (from K^2 to \mathbb{R}) as*

$$\begin{aligned} Q_0(x, x') &= \sigma_b^2 + \frac{\sigma_w^2}{p} x \cdot x'; \\ Q_l &= \alpha Q_{l-1} + \beta \left(\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}} \right) Q_{l-1} \right); \\ C_l(x, x') &= \frac{Q_l(x, x')}{\sqrt{Q_l(x, x)Q_l(x', x')}} \end{aligned}$$

where $\alpha \geq 0$ and $\beta > 0$. For any l , Q_l and C_l are kernels on K , in the sense of Definition 1.

Proof. It is straightforward to prove that Q_0 is a kernel. Now let us show that if Q_l is a kernel for some l , then C_l is a kernel. Since Q_l is symmetric, so is C_l . Moreover, one can easily check that the diagonal elements of Q_l are continuous and do not vanish. Hence C_l is continuous. Moreover, the non-negative definiteness of $T_\mu(Q_l)$ implies that $T_\mu(C_l)$ is non-negative definite for any finite Borel measure μ , and so C_l is a kernel if Q_l is.

We proceed by induction and assume that Q_{l-1} and C_{l-1} are kernels. Then $f(C_{l-1})$ is a kernel as well, by Lemmas 8 and 9, and $Q_{l-1}/C_{l-1} : (x, x') \mapsto \sqrt{Q_{l-1}(x, x)Q_{l-1}(x', x')}$ is also a kernel. We conclude using the fact that the sum of two kernels is a kernel, and multiplying a kernel for a positive constant also gives a kernel (Paulsen and Raghupathi, 2016). \square

Proposition 1. *Fixed any compact $K \subset \mathbb{R}^p$, for $l \in [1 : L]$, Q_l is a universal kernel on K .*

Proof. Fix $l \in [1 : L]$. From Lemma 8 and Lemma 9, we know that, $f(C_{l-1})$ is a kernel. Moreover, using the fact that finite sum and point-wise multiplication of kernels are kernels (Paulsen and Raghupathi, 2016), we obtain that $\sigma_b^2 + \frac{\sigma_w^2}{2} \left(1 + \frac{f(C_{l-1})}{C_{l-1}}\right) Q_{l-1}$ is a kernel.

Now, notice that given two kernels, Q and Q' , their sum is a kernel whose RKHS satisfies $\mathcal{H}_{Q+Q'} \supseteq \mathcal{H}_Q$ and $\mathcal{H}_{Q+Q'} \supseteq \mathcal{H}_{Q'}$ (Paulsen and Raghupathi, 2016). Moreover, for any kernel Q , for all $\alpha > 0$, αQ is a kernel, and its RKHS coincides with the one of Q (up to a rescaling of the norm). We hence conclude from (2.7) that $\mathcal{H}_{Q_l} \supseteq \mathcal{H}_{Q_{l-1}}$. Thus, if we can show that Q_1 is universal, we also get the universality of Q_l . From Proposition 2 (proved in the next section), we have that $C_0 + f(C_0)$ is universal. In particular, by Lemma 3, for all non-zero finite Borel measure μ on K , $T_\mu(C_0 + f(C_0))$ is strictly positive definite. Define $R_0 = Q_0/C_0$. For all $x, x' \in K$, we have $R_0(x, x') = \sqrt{Q_0(x, x)Q_0(x', x')}$, which is easy to check to define a kernel.

Now, let us show that $R_0(C_0 + f(C_0))$ is universal by Lemma 3. Indeed, for any non-zero finite Borel measure μ on K , we have that $T_\mu(R_0(C_0 + f(C_0)))$ is strictly positive definite, since $\langle T_\mu(R_0(C_0 + f(C_0))) \varphi, \varphi \rangle = \langle T_\mu(C_0 + f(C_0)) \psi, \psi \rangle$ for all $\varphi \in L^2(K, \mu)$, where $\psi(x) = \sqrt{Q_0(x, x)} \varphi(x)$. We conclude by noticing that $Q_1 - \frac{\sigma_w^2}{2} R_0(C_0 + f(C_0)) = \sigma_b^2$ is a kernel, and so the RKHS of Q_1 must contain the one of $R_0(C_0 + f(C_0))$. \square

2.6.3 Toy model

We recall that $I \subset (0, \infty)$ is a compact interval, and ρ is the standard Lebesgue measure on I . Since ρ will be the only measure involved in the whole discussion on the toy model, we will omit its explicit dependence. Moreover, to simplify the notations, we will make no distinctions of notations between kernels on I and their induced operators, namely we will denote as Q the integral operator $T_\rho(Q)$ induced by a kernel Q .

Proof of Proposition 3

Lemma 11. *Let $L^2(I) = \mathcal{H}_0 \oplus \mathcal{H}_1$, where $\mathcal{H}_0 = \text{Span}(1, \text{id})$ in $L^2(I)$, with the notation $1 : x \mapsto 1$ and $\text{id} : x \mapsto x$. For all $t \in [0, 1]$ we can write a decomposition of q_t as*

$$q_t = q_t^0 \oplus kt^{5/2}(E + B_t)$$

where k is a real positive constant, $q_t^0 : \mathcal{H}_0 \rightarrow \mathcal{H}_0$, $E : \mathcal{H}_1 \rightarrow \mathcal{H}_1$, $B_t : \mathcal{H}_1 \rightarrow \mathcal{H}_1$, and the following properties hold:

- q_t^0 is non-negative definite for all $t \in [0, 1]$ and there exists a $T \in (0, 1)$ such that q_t^0 is strictly positive for all $t \in (0, T)$;
- E is strictly positive definite;
- $\lim_{t \rightarrow 0} \|B_t\|_2 = 0$.

Proof. Recall that $f(c) = \frac{1}{\pi} \left(\sqrt{1-c^2} - c \arccos c \right)$ can be expanded around 1 as

$$f(c) = \frac{2\sqrt{2}}{3\pi} (1-c)^{3/2} + O\left((1-c)^{5/2}\right).$$

This allows us to write an expansion of q_t around $t = 0$. For any $(x, y) \in I^2$ and $t > 0$

$$q_t(x, y) = e^{\sigma_w^2 t/2} xy + \frac{2\sigma_w^2}{\sigma_b^2} \left(e^{\sigma_w^2 t/2} - 1 \right) + \frac{\sigma_b^3 \sigma_w^2}{15\pi} \frac{|x-y|^3}{(xy)^2} t^{5/2} + o(t^{5/2}).$$

Let us define the integral operators \hat{q}_t^0 and \hat{E} on I , via

$$\begin{aligned} \hat{q}_t^0(x, y) &= e^{\sigma_w^2 t/2} xy + \frac{2\sigma_w^2}{\sigma_b^2} \left(e^{\sigma_w^2 t/2} - 1 \right); \\ \hat{E}(x, y) &= \frac{|x-y|^3}{(xy)^2}. \end{aligned} \tag{2.22}$$

We can then write,

$$q_t = \hat{q}_t^0 + k t^{5/2} (\hat{E} + \hat{B}_t),$$

where $k > 0$ and for all (x, y) , $\hat{B}_t(x, y) \rightarrow 0$ by definition. Moreover, $t \mapsto \hat{B}_t$ is a continuous map with respect to the L^2 operator norm, since q_t is continuous (by a result equivalent to Corollary 1). Hence $\lim_{t \rightarrow 0} \|\hat{B}_t\|_2 = 0$. \hat{q}_t^0 is supported in \mathcal{H}_0 and have range $\mathcal{R}(\hat{q}_t^0) \subseteq \mathcal{H}_0$, so it is well defined as an operator $\mathcal{H}_0 \rightarrow \mathcal{H}_0$. Furthermore, \hat{q}_t^0 is strictly positive definite, on \mathcal{H}_0 , for all $t \in (0, 1]$.

However, we cannot conclude yet, since \hat{E} and \hat{B}_t are supported on the whole $L^2(I)$ and \hat{E} is not positive definite. Let us denote as P_0 and P_1 the projectors on \mathcal{H}_0 and \mathcal{H}_1 respectively.

Define $q_t^0 : \mathcal{H}_0 \rightarrow \mathcal{H}_0$, $E : \mathcal{H}_1 \rightarrow \mathcal{H}_1$, $B_t : \mathcal{H}_1 \rightarrow \mathcal{H}_1$ as

$$q_t^0 = P_0 q_t P_0 = \hat{q}_t^0 + k t^{5/2} P_0 (\hat{E} + \hat{B}_t) P_0; \quad E = P_1 \hat{E} P_1; \quad B_t = P_1 \hat{B}_t P_1.$$

Clearly $q_t = q_t^0 \oplus k t^{5/2} (E + B_t)$. We now need to check that the decomposition satisfies the required properties. First, notice that $\|B_t\|_2 \leq \|\hat{B}_t\|_2 \rightarrow 0$, so $\|B_t\|_2 \rightarrow 0$ for $t \rightarrow 0$. Let us now focus on E . By Lemma 12 (proved in a later section), \hat{E} is compact and self-adjoint. The same holds for E since it is (the restriction of) a conjugate of \hat{E} under a compact self-adjoint projector. Now consider $\varphi \in \mathcal{H}_1$ such that $E\varphi = 0$. We have that $\hat{E}\varphi(x) = \alpha + \beta x$ for some real α and β . Now, applying the left inverse \hat{F} , defined in Lemma 12, we get $\varphi = \hat{F}\hat{E}\varphi = 0$ and thus E is injective. We need to show that E is strictly positive definite, which is true if and only if all its eigenvalues are strictly positive, by the spectral theorem for compact self-adjoint operators (Lang, 2012). Let $\varphi \in \mathcal{H}_1$ be a normalized eigenfunction of E with eigenvalue λ . Using the fact that q_t is non-negative definite (by an equivalent of Lemma 6), we have for all $t > 0$

$$\langle q_t \varphi, \varphi \rangle = k t^{5/2} \langle (E + B_t) \varphi, \varphi \rangle = k t^{5/2} (\lambda + \langle B_t \varphi, \varphi \rangle) \geq 0.$$

Since $\langle B_t \varphi, \varphi \rangle \rightarrow 0$ for $t \rightarrow 0$, we conclude that $\lambda \geq 0$. The previously proven injectivity of E shows that $\lambda > 0$.

Finally, as for q_t^0 , it is straightforward to see that it is non-negative definite since q_t is. On the (non orthogonal) basis $\{1, \text{id}\}$ of \mathcal{H}_0 , q_t^0 is represented by a matrix in the form

$$\begin{pmatrix} e^{\frac{\sigma_w^2}{2}} & 0 \\ 0 & \frac{\sigma_w^4}{\sigma_b^2} t \end{pmatrix} + o(t),$$

which has rank 2 for $t > 0$ small enough. Hence the asymptotic strict positivity follows. \square

Lemma 12. *The operator \hat{E} defined by (2.22) induces a compact self-adjoint operator on $L^2(I)$. The operator \hat{F} given by $\hat{F}\varphi(x) = x^2 \frac{d^4}{dx^4}(x^2\varphi(x))$ is well defined on the range $\mathcal{R}(\hat{E})$ and is a left inverse of \hat{E} on the whole $L^2(I)$.*

Proposition 3. *For any non-zero $\varphi \in L^2(I, \rho)$, there is a $t_\varphi \in (0, 1]$ such that $\langle q_t \varphi, \varphi \rangle > 0$, for all $t \in (0, t_\varphi)$.*

Proof. With the notation of Lemma 11, it is enough to show that the statement holds for $\varphi \in \mathcal{H}_0$ and $\varphi \in \mathcal{H}_1$. For $\varphi \in \mathcal{H}_0$, it is a straight consequence of the strict positivity of q_t^0 for $t > 0$ small enough. Now, fix a non-zero $\varphi \in \mathcal{H}_1$ and let $p = \langle E\varphi, \varphi \rangle > 0$. Define $\varepsilon : [0, 1] \rightarrow \mathbb{R}$ as $\varepsilon(t) = \sup_{s \in [0, t]} \|B_s\|_2$. By definition, ε is non-increasing and vanishes for $t \rightarrow 0$. Then, there exists a $t_\varphi > 0$ such that $p > \varepsilon(t_\varphi) \|\varphi\|_2^2$, for all $t \in [0, t_\varphi)$. It follows that $\langle (E + B_t)\varphi, \varphi \rangle > 0$, and so $\langle q_t \varphi, \varphi \rangle > 0$ for all $t \in (0, t_\varphi)$. \square

Proposition 4. *For all $\varphi \in L^2(I, \rho)$, the map $t \mapsto \langle q_t \varphi, \varphi \rangle$ is non decreasing on $[0, 1]$.*

Proof. It is enough to show that for $t \in [0, 1]$, \dot{q}_t is a non-negative definite operator, and then conclude with the same argument we will use in the proof of Corollary 1. Notice that

$$\dot{q}_t = \sigma_b^2 + \frac{\sigma_w^2}{2} q_t + \frac{\sigma_w^2}{2} \frac{f(c_t)}{c_t} q_t,$$

also holds with the derivative taken with respect to the operator norm (by an equivalent of Corollary 1).

Now fix $\varphi \in L^2(I)$. Since q_t is non-negative (by an equivalent of Lemma 6), we can write

$$\begin{aligned} \langle \dot{q}_t \varphi, \varphi \rangle &= \sigma_b^2 |\langle 1, \varphi \rangle|^2 + \frac{\sigma_w^2}{2} \langle q_t \varphi, \varphi \rangle + \frac{\sigma_w^2}{2} \left\langle \frac{f(c_t)}{c_t} \circ q_t \varphi, \varphi \right\rangle \\ &= \sigma_b^2 |\langle 1, \varphi \rangle|^2 + \frac{\sigma_w^2}{4} \langle q_t \varphi, \varphi \rangle + \frac{\sigma_w^2}{2} \left\langle \frac{c_t/2 + f(c_t)}{c_t} \circ q_t \varphi, \varphi \right\rangle \\ &\geq \frac{\sigma_w^2}{2} \left\langle \frac{g(c_t)}{c_t} \circ q_t \varphi, \varphi \right\rangle, \end{aligned}$$

where $g(c_t)$ is the integral operator defined by $g(c_t)(x, y) = c_t(x, y)/2 + f(c_t(x, y))$.

With a slight abuse of notation we denote with g the map $[-1, 1] \rightarrow \mathbb{R}$ given by $g(z) = z/2 + f(z)$. It is not hard to prove (see Lemma A8 in Hayou et al. (2021)) that the Taylor expansion of g around 0 is convergent on $[-1, 1]$, and all its coefficients are non-negative. Hence Lemma 8 applies, and $g(c_t)$ is non-negative definite.

To conclude it is enough to notice that $\frac{g(c_t)}{c_t} \circ q_t = g(c_t) \circ \frac{q_t}{c_t}$. By definition of c_t , $\frac{q_t}{c_t}(x, y) = \sqrt{q_t(x, x) q_t(y, y)}$. Hence we have

$$\langle \dot{q}_t \varphi, \varphi \rangle \geq \frac{\sigma_w^2}{2} \langle g(c_t) \psi, \psi \rangle \geq 0,$$

where $\psi(x) = \sqrt{q_t(x, x)} \varphi(x)$. \square

Lemma 12. *The operator \hat{E} defined by (2.22) is a compact self-adjoint operator on $L^2(I)$. The operator \hat{F} given by $\hat{F}\varphi(x) = x^2 \frac{d^4}{dx^4}(x^2\varphi(x))$ is well defined on $\mathcal{R}(\hat{E})$ and is a left inverse of \hat{E} on the whole $L^2(I)$.*

Proof. \hat{E} is clearly compact and self-adjoint since $(x, y) \mapsto \frac{|x-y|^3}{x^2y^2}$ is a continuous real symmetric map on I^2 .

Let Λ denote the $L^2(I)$ isomorphism $\Lambda\varphi(x) \mapsto \varphi(x)/x^2$. With the notations of Lemma 13, we have $\hat{E} = \Lambda E_3 \Lambda$ and $\hat{F} = \Lambda^{-1} F_3 \Lambda^{-1}$. By Lemma 13, we conclude. \square

Lemma 13. *For $n \in \mathbb{N}$, let E_n be the integral operator on $L^2(I)$ defined via*

$$E_n \varphi : x \mapsto \int_I (x-y)^n \text{sign}(x-y) \varphi(y) dy.$$

The following properties hold:

- (a) *The range $\mathcal{R}(E_n)$ is contained in $C^n(I)$. If $n \geq q$, for any $\varphi \in L^2(I)$, denoting $\psi = E_n \varphi \in C^n(I)$, for $k \in \{1 \dots n\}$, ψ 's k -th derivative is given by*

$$\psi^{(k)} = \frac{n!}{(n-k)!} E_{n-k} \varphi.$$

- (b) *For all $\varphi \in L^2(I)$, $\psi = E_0 \varphi$ is absolutely continuous and $\psi' = 2\varphi$. As a consequence $\mathcal{R}(E_n) \subseteq H^{n+1}(I) = W^{n+1,2}(I)$.*

- (c) *On the whole $L^2(I)$, E_n admits a left inverse F_n , defined on $H^{n+1}(I)$ as*

$$F_n \varphi = \frac{1}{2n!} \varphi^{(n+1)}.$$

- (d) *E_n is injective.*

Proof. Let $I = [A, B]$ and notice that the action of E_n can be rewritten as

$$E_n \varphi(x) = \int_A^x (x-y)^n \varphi(y) dy + \int_B^x (x-y)^n \varphi(y) dy.$$

(a) For $n \geq 1$, since $z \mapsto z^n \text{sign } z$ is C^{n-1} , standard tools from functional analysis show that $\psi = E_n \varphi \in C^{n-1}$ and $\psi^{(k)} = \frac{n!}{(n-k)!} E_{n-k} \varphi$ for $k \in \{1 \dots n-1\}$. Then, to show that E_n

has range in C^n , it is sufficient to prove that $E_1 \varphi$ is C^1 for all squared integrable φ . This can be done explicitly from the definition of derivative since

$$E_1 \varphi(x + \varepsilon) - E_1 \varphi(x) = \varepsilon E_0 \varphi(x + \varepsilon) + 2 \operatorname{sign} \varepsilon \int_x^{x+\varepsilon} (y - x) \varphi(y) dy.$$

E_0 is continuous since both $\int_A^x \varphi(y) dy$ and $\int_B^x \varphi(y) dy$ are, and by Cauchy-Schwartz the last term is bounded by $2 \|\varphi\|_2 \sqrt{\varepsilon^3/3} = o(\varepsilon)$.

(b) The first statement is a straight consequence of Lebesgue's differentiation theorem. Then, applying (a), we get that the n -th derivative of ψ is absolutely continuous and so $\psi \in H^{n+1}(I)$.

(c) It follows directly from (a) and (b).

(d) For all φ in $L^2(I)$, $E_n \varphi = 0$ implies $\varphi = F_n E_n \varphi = 0$. \square

2.6.4 Continuous limit

Lemma 6. *For any x, x' in K , the solution of (2.18) is unique and well defined for all $t \in [0, 1]$. The maps $(x, x') \mapsto q_t(x, x')$ and $(x, x') \mapsto c_t(x, x')$ are Lipschitz continuous on K^2 and c_t takes values in $[-1, 1]$. Moreover, both q_t and c_t are kernels in the sense of Definition 1.*

Proof. First, notice that from (2.18) we can find, with few algebraic manipulations, an explicit recurrence relation for the correlation C_l , defined in (2.5). For any $x, x' \in K$ we have

$$C_{l+1}(x, x') = A_{l+1}(x, x') C_l(x, x') + \frac{\sigma_w^2}{2L} \left(1 + \frac{\sigma_w^2}{2L}\right)^{-1} A_{l+1}(x, x') f(C_l(x, x')) + \frac{1}{L} \frac{\sigma_b^2}{\sqrt{Q_l(x, x) Q_l(x', x')}}; \quad (2.23)$$

$$A_l(x, x') = \sqrt{\left(1 - \frac{1}{L} \frac{\sigma_b^2}{Q_l(x, x)}\right) \left(1 - \frac{1}{L} \frac{\sigma_b^2}{Q_l(x', x')}\right)}.$$

We can find a Cauchy problem for the correlation directly from (2.18), or by noting that $A_l(x, x') = 1 - \frac{\sigma_b^2}{2L} \left(\frac{1}{Q_l(x, x)} + \frac{1}{Q_l(x', x')}\right) + o(1/L)$, for $L \rightarrow \infty$. With both approaches, we have

$$\begin{aligned} \dot{c}_t(x, x') &= \sigma_b^2 (\mathcal{G}_t(x, x') - \mathcal{A}_t(x, x') c_t(x, x')) + \frac{\sigma_w^2}{2} f(c_t(x, x')), \\ c_0(x, x') &= \frac{\sigma_b^2 + \sigma_w^2 x \cdot x'}{\sqrt{(\sigma_b^2 + \sigma_w^2 \|x\|^2)(\sigma_b^2 + \sigma_w^2 \|x'\|^2)}}, \end{aligned} \quad (2.24)$$

where f is defined in (2.6) and

$$\mathcal{A}_t(x, x') = \frac{1}{2} \left(\frac{1}{q_t(x, x)} + \frac{1}{q_t(x', x')} \right); \quad \mathcal{G}_t(x, x') = \sqrt{\frac{1}{q_t(x, x) q_t(x', x')}}.$$

Note that for the diagonal terms $q_t(x, x)$, (2.18) reduces to $\dot{q}_t = \sigma_b^2 + \frac{\sigma_w^2}{2} q_t$, whose solution is

$$q_t(x, x) = e^{\frac{\sigma_w^2}{2} t} q_0(x, x) + \frac{2\sigma_b^2}{\sigma_w^2} \left(e^{\frac{\sigma_w^2}{2} t} - 1 \right) = e^{\frac{\sigma_w^2}{2} t} (\sigma_b^2 + \sigma_w^2 \|x\|^2) + \frac{2\sigma_b^2}{\sigma_w^2} \left(e^{\frac{\sigma_w^2}{2} t} - 1 \right).$$

Now, fix $z = (x, x') \in K^2$ and let $\gamma_0 = c_0(z) \in [-1, 1]$. Consider $\bar{f} : \mathbb{R} \rightarrow \mathbb{R}$, an arbitrary Lipschitz extension of f to the whole \mathbb{R} and define $H : [0, \infty) \times \mathbb{R} \rightarrow \mathbb{R}$ as

$$H(t, \gamma) = \sigma_b^2 (\mathcal{G}_t(z) - \mathcal{A}_t(z) \gamma) + \frac{\sigma_w^2}{2} \bar{f}(\gamma).$$

H is Lipschitz continuous in γ and C^∞ in t , so there exists $\tau > 0$ such that the Cauchy problem

$$\dot{\gamma}(t) = H(t, \gamma(t)); \quad \gamma(0) = \gamma_0$$

has a unique C^1 solution defined for $t \in [0, \tau)$. Noticing that

$$\mathcal{G}_t(x, x') - \mathcal{A}_t(x, x') = -\frac{1}{2} \left(\frac{1}{q_t(x, x)} - \frac{1}{q_t(x', x')} \right)^2 \leq 0,$$

we get that for all t_1 such that $\gamma(t_1) = 1$ we have $\dot{\gamma}(t_1) \leq 0$, since $f(1) = 0$, and for all t_{-1} such that $\gamma(t_{-1}) = -1$ we have $\dot{\gamma}(t_{-1}) = \sigma_b^2 (\mathcal{G}_t(x, x') + \mathcal{A}_t(x, x')) + \frac{\sigma_w^2}{2} > 0$. As a consequence $\gamma(t) \in [-1, 1]$ for all $t \in [0, \tau)$ and we can take $\tau = \infty$. In particular we get that (2.24) has a unique solution $t \mapsto c_t(z)$, defined for $t \in [0, 1]$ and bounded in $[-1, 1]$. Consequently, (2.18) has a unique and well-defined solution for all $t \geq 0$.

Now notice that $z \mapsto c_0(z)$ is Lipschitz on K^2 . let us denote as L_0 a Lipschitz constant for c_0 . Since both \mathcal{G}_t and \mathcal{A}_t are C^1 , we can find real constants L_G , L_A and M_A such that for all z, z' elements of K^2

$$|\mathcal{G}_t(z) - \mathcal{G}_t(z')| \leq L_G \|z - z'\|; \quad |\mathcal{A}_t(z) - \mathcal{A}_t(z')| \leq L_A \|z - z'\|; \quad |\mathcal{A}_t(z)| \leq M_A.$$

Let L_f be a Lipschitz constant for f . Using the fact that $|c_t| \leq 1$, we can write

$$|\dot{c}_t(z) - \dot{c}_t(z')| \leq L_1 \|z - z'\| + L_2 |c_t(z) - c_t(z')|,$$

where $L_1 = \sigma_b^2(L_G + L_A)$ and $L_2 = \sigma_b^2 M_A + \frac{\sigma_w^2}{2} L_f$. Now fix z and z' and consider $\Delta(t) = c_t(z) - c_t(z')$. We have

$$|\dot{\Delta}(t)| \leq L_1 \|z - z'\| + L_2 |\Delta(t)|; \quad |\Delta(0)| \leq L_0 \|z - z'\|.$$

So,

$$|\Delta(t)| \leq \left(\frac{L_1}{L_2} (e^{L_2 t} - 1) + L_0 e^{L_2 t} \right) \|z - z'\|,$$

meaning that c_t (and so q_t) is Lipschitz on L^2 .

Since the mapping $(x, x') \mapsto q_t(x, x')$ is continuous, it defines a compact integral operator $T(q_t)$ on $L^2(K)$ (Lang, 2012). Since q_t is real and symmetric under the swap of x and x' , the operator is self-adjoint. The same holds for c_t . The fact that $T(q_t)$ is a non-negative operator can be seen as a corollary of Lemma 7. Indeed, since a kernel induces it, every $T(Q_{l|L})$ is a non-negative definite operator. Hence, for each $t \in [0, 1]$ it is enough to find a sequence $\{l_n, L_n\}_{n \in \mathbb{N}}$ (where $L_n \geq 1$ is an integer and $l_n \in [0 : L_n]$) such that $L_n \rightarrow \infty$ and $l_n/L_n \rightarrow t$. By Lemma 7, $T(Q_{l_n|L_n}) \rightarrow T(q_t)$ in the L^∞ norm, and hence in L^2 , as we are on a compact set. By Lemma 10, for all $n \in \mathbb{N}$ we have that $T(Q_{l_n|L_n})$ is non-negative definite. Since the subspace of non-negative definite operators in L^2 is closed with respect to the L^2 operator norm, we conclude. Now that we have established that $T(q_t)$ is non-negative definite, it follows immediately that $T(c_t)$ is also non-negative. Since these results hold for any arbitrary finite Borel measure μ on K , we can thus conclude by Lemma 1 that both q_t and c_t are kernels, in the sense of Definition 1. \square

Corollary 1. *Fix any finite Borel measure μ on K and recall the notation T_μ , introduced in Section 2.3.1, for the integral operator induced by a kernel. The maps $t \mapsto T_\mu(q_t)$ and $t \mapsto T_\mu(c_t)$, defined on $[0, 1]$, are continuous and twice differentiable with respect to the operator norm in $L^2(K, \mu)$. Moreover, $\frac{d}{dt} T_\mu(q_t) = T_\mu(\dot{q}_t)$, $\frac{d}{dt} T_\mu(c_t) = T_\mu(\dot{c}_t)$, $\frac{d^2}{dt^2} T_\mu(q_t) = T_\mu(\ddot{q}_t)$ and $\frac{d^2}{dt^2} T_\mu(c_t) = T_\mu(\ddot{c}_t)$.*

Proof. Consider the map $(t, z) \mapsto q_t(z)$, defined on $[0, 1] \times K^2$, which is continuous with respect

to z and C^2 with respect to t , as one can easily check. Since K^2 and $[0, 1]$ are compact sets, it follows that for any t

$$\limsup_{s \rightarrow t} \sup_{z \in I^2} \left| \frac{q_s(z) - q_t(z)}{s - t} - \dot{q}_t(z) \right| = \sup_{z \in I^2} \lim_{s \rightarrow t} \left| \frac{q_s(z) - q_t(z)}{s - t} - \dot{q}_t(z) \right| = 0.$$

Hence $\lim_{s \rightarrow t} \frac{q_s - q_t}{t - s} = \dot{q}_t$ uniformly on K^2 , and hence $\lim_{s \rightarrow t} \frac{T_\mu(q_s) - T_\mu(q_t)}{t - s} = T_\mu(\dot{q}_t)$ in the $L^2(K, \mu)$ norm for operators, since K is compact. The proof for the second derivative works similarly, using the fact that $(t, z) \mapsto q_t(z)$ is continuous in z and C^1 in t . As a consequence of the above results, $t \mapsto T_\mu(q_t)$ is continuous and twice differentiable, with $\frac{d}{dt} T_\mu(q_t) = T_\mu(\dot{q}_t)$ and $\frac{d^2}{dt^2} T_\mu(q_t) = T_\mu(\ddot{q}_t)$.

The proof for $T_\mu(c_t)$ is analogous. \square

Lemma 7 (Convergence to the continuous limit). *Let $Q_{l|L}$ be the covariance function of the layer l in a net of $L + 1$ layers $[0 : L]$, and q_t be the solution of (2.18), then*

$$\lim_{L \rightarrow \infty} \sup_{l \in [0:L]} \sup_{(x, x') \in K^2} |Q_{l|L}(x, x') - q_{t=l/L}(x, x')| = 0.$$

Proof. We will show that the relation holds for c_t and hence for q_t . Let H , defined on $[0, 1] \times K^2$, be such that $\dot{c}_t(z) = H(z, t, c_t(z))$. Explicitly, with the same notations as in (2.24),

$$H(z, t, \gamma) = \sigma_b^2 (\mathcal{G}_t(z) - \mathcal{A}_t(z) \gamma) + \frac{\sigma_w^2}{2} f(\gamma).$$

Define

$$\tau(h) = \sup_{t, z} \left| \frac{c_{t+h}(z) - c_t(z)}{h} - H(z, t, c_t(z)) \right|.$$

Since t and z take values on compact sets, by uniform continuity, we can write

$$\sup_t \sup_{s \in [t, t+h]} |H(z, s, c_s(z)) - H(z, t, c_t(z))| = o(h),$$

as $h \rightarrow 0$. Since τ can be written as $\tau(h) = \frac{1}{h} \sup_{t, z} \left| \int_t^{t+h} (H(z, s, c_s(z)) - H(z, t, c_t(z))) ds \right|$, it is clear that $\tau(h) \rightarrow 0$ for $h \rightarrow 0$. Now, for any integer $L \geq 1$, let $\tilde{H}_L : K^2 \times [0 : L - 1] \times [-1, 1]$

be given by

$$\begin{aligned} \tilde{H}_L(z, l, \gamma) = & (A_{l+1|L}(x, x') - 1) L \gamma + \frac{\sigma_w^2}{2} \left(1 + \frac{\sigma_w^2}{2L}\right)^{-1} A_{l+1|L}(x, x') f(c_l(x, x')) \\ & + \frac{\sigma_b^2}{\sqrt{Q_{l|L}(x, x) Q_{l|L}(x', x')}} , \end{aligned}$$

where,

$$A_{l|L}(x, x') = \sqrt{\left(1 - \frac{1}{L} \frac{\sigma_b^2}{Q_{l|L}(x, x)}\right) \left(1 - \frac{1}{L} \frac{\sigma_b^2}{Q_{l|L}(x', x')}\right)}.$$

It is clear from (2.23) that \tilde{H}_L has been defined so that $C_{l+1|L}(z) - C_{l|L}(z) = \frac{1}{L} \tilde{H}_L(z, l, \gamma)$, for all $L \in [0 : L - 1]$ and all $z \in K^2$. Using the explicit form of the diagonal terms of Q and q , it can be easily shown that, for $L \rightarrow \infty$,

$$\begin{aligned} \sup_{(x, x') \in K^2} \sup_{l \in [0 : L - 1]} A_{l+1|L}(x, x') &= 1 + \frac{\sigma_b^2}{L} \mathcal{A}_{t=l/L}(x, x') + O(1/L^2); \\ \sup_{(x, x') \in K^2} \sup_{l \in [0 : L]} \frac{\sigma_b^2}{\sqrt{Q_{l|L}(x, x) Q_{l|L}(x', x')}} &= \mathcal{G}_{t=l/L}(x, x') + O(1/L^2), \end{aligned}$$

where \mathcal{A}_t and \mathcal{G}_t are defined as in (2.24). As a consequence, we can find a constant $M_1 > 0$ and an integer $L_\star > 0$ such that, for all $\gamma \in [-1, 1]$, for all $z \in K^2$, for all $L \geq L_\star$

$$|\tilde{H}_L(z, l, \gamma) - H(z, l/L, \gamma)| \leq \frac{M_1}{L}. \quad (2.25)$$

Moreover, there exists a constant $M_2 > 0$ such that for all $z \in K^2$, all $t \in [0, 1]$ and all pairs $(\gamma, \gamma') \in [-1, 1]^2$

$$|H(z, t, \gamma) - H(z, t, \gamma')| \leq M_2 \|\gamma - \gamma'\|. \quad (2.26)$$

Thanks to the two above uniform inequalities, we will now show that, for $L \geq L_\star$,

$$\sup_{l \in [0 : L]} \sup_{z \in K^2} |C_{l|L}(x, x') - c_{t(l/L)}(x, x')| \leq \tilde{\tau}(1/L) \frac{e^{M_2} - 1}{M_2}, \quad (2.27)$$

where $\tilde{\tau} : h \mapsto \tau(h) + M_1 h$. To do so, fix $L \geq L_\star$ and define $\Delta_l = \sup_{z \in K^2} |C_{l|L}(x, x') -$

$c_{t(l,L)}(x, x')$. Using the definition of τ , (2.25) and (2.26) we get

$$|\Delta_{l+1}| \leq \left(1 + \frac{M_2}{L}\right) |\Delta_l| + \frac{1}{L} \tau(1/L) + \frac{M_1}{L} = \left(1 + \frac{M_2}{L}\right) |\Delta_l| + \frac{1}{L} \tilde{\tau}(1/L).$$

At this point, using the fact that $\Delta_0 = 0$, it is easy to show by induction that

$$\Delta_l \leq \tilde{\tau}(1/L) \frac{\left(1 + \frac{M_2}{L}\right)^l - 1}{M_2},$$

and so (2.27) follows. Finally, the uniform convergence of C to c implies the one of Q to q , so we conclude. \square

2.6.5 Universality for the uniform scaling

To prove Theorem 6, the idea is to prove that for any finite Borel measure μ on K , the operator $T_\mu(q_t)$ is strictly positive definite if $t > 0$, and then use the characterization of universal kernels from Lemma 3. To prove the strict positive definiteness, we will proceed in two steps. First we show in Proposition 9 that, for all non-zero $\varphi \in L^2(K, \mu)$, $\langle T_\mu(q_t) \varphi, \varphi \rangle > 0$ for t small enough. Then we use Proposition 10, which shows that $\frac{d}{dt} T_\mu(q_t)$ is non-negative definite.

Proposition 9. *Fix any finite Borel measure μ on K , and assume that $\sigma_b > 0$. Given any non-zero $\varphi \in L^2(K, \mu)$, there exists a $t_\varphi \in (0, 1]$ such that $\langle T_\mu(q_t) \varphi, \varphi \rangle > 0$, for all $t \in (0, t_\varphi)$.*

Proof. From Corollary 1, we can expand $T_\mu(q_t)$ around $t = 0$ as

$$T_\mu(q_t) = T_\mu(q_0) + t T_\mu(\dot{q}_0) + o(t) = t T_\mu \left(\sigma_b^2 + \frac{\sigma_w^2}{2} q_0 \right) + T_\mu((c_0 + t f(c_0)) R_0) + o(t),$$

$o(t)$ being with respect to the operator norm, where we have defined the kernel R_0 via $R_0(x, x') = \frac{\sigma_w^2}{2} \sqrt{(1 + \zeta \|x\|^2)(1 + \zeta \|x'\|^2)}$. Since $T_\mu(q_0)$ is non-negative, for any $\varphi \in L^2(I)$, we have

$$\begin{aligned} \langle T_\mu(q_t) \varphi, \varphi \rangle &\geq \langle T_\mu((c_0 + t f(c_0)) R_0) \varphi, \varphi \rangle + o(t) = \left(1 - \frac{t}{2}\right) \langle T_\mu(c_0) \psi, \psi \rangle \\ &\quad + t \langle T_\mu(f(c_0)) \psi, \psi \rangle + o(t), \end{aligned}$$

where $\psi(x) = \sigma_w \sqrt{(1 + \zeta \|x\|^2)/2} \varphi(x)$. We conclude by the strict positivity of $\tilde{f}(c_0)$ on

$L^2(K, \mu)$, thanks to Proposition 2 and Lemma 3. \square

Proposition 10. *For any finite Borel measure μ on K , for any $t \in [0, 1]$, the operator $T_\mu(\dot{q}_t)$ on $L^2(K, \mu)$ is non-negative definite. In particular, for all $\varphi \in L^2(K, \mu)$ we have*

$$\frac{d}{dt} \langle T_\mu(q_t) \varphi, \varphi \rangle \geq 0.$$

Proof. Fix μ and $\varphi \in L^2(K, \mu)$. From (2.18) we can write

$$T_\mu(\dot{q}_t) = T_\mu \left(\sigma_b^2 + \frac{\sigma_w^2}{2} q_t + \frac{\sigma_w^2}{2} \frac{f(c_t)}{c_t} q_t \right).$$

By Lemma 6, $T_\mu(q_t)$ is non-negative definite, so we can write

$$\begin{aligned} \langle T_\mu(\dot{q}_t) \varphi, \varphi \rangle &= \sigma_b^2 |\langle 1, \varphi \rangle|^2 + \frac{\sigma_w^2}{2} \left\langle T_\mu \left(\frac{c_t + f(c_t)}{c_t} q_t \right) \varphi, \varphi \right\rangle \\ &\geq \frac{\sigma_w^2}{2} \left\langle T_\mu \left(\tilde{f}(c_t) \frac{q_t}{c_t} \right) \varphi, \varphi \right\rangle = \frac{\sigma_b^2}{2} \langle T_\mu(\tilde{f}(c_t)) \psi, \psi \rangle, \end{aligned}$$

where $\tilde{f} : \gamma \mapsto \frac{\gamma}{2} + f(\gamma)$, for $\gamma \in [-1, 1]$, and $\psi(x) = \sqrt{q_t(x, x)} \varphi(x)$. By Lemma 9, the Taylor expansion of \tilde{f} around 0 converges uniformly on $[-1, 1]$, and all its coefficients are non-negative. We conclude by Lemma 8 that $T_\mu(\dot{q}_t)$ is non-negative definite. Finally, to prove the inequality, it is enough to recall that $\frac{d}{dt} T_\mu(q_t) = T_\mu(\dot{q}_t)$ by Corollary 1, the derivative $\frac{d}{dt}$ being with respect to the operator norm on $L^2(K, \mu)$. \square

Theorem 6 (Universality of q_t). *Let $K \subset \mathbb{R}^p$ be compact. For any $t \in (0, 1]$, the solution q_t of (2.18) is a universal kernel on K .*

Proof. By Lemma 3, it suffices to show that for any finite Borel measure μ on K , $T_\mu(q_t)$ is strictly positive definite for all $t \in (0, 1]$. Fix any nonzero $\varphi \in L^2(K, \mu)$, define the map F on $[0, 1]$ by $F(t) = \langle T_\mu(q_t) \varphi, \varphi \rangle$. For any fixed $t \in (0, 1]$, by Proposition 9 we can find $s \in (0, t)$ such that $F(s) > 0$. Since F is non decreasing by Proposition 10, we get that $F_t > 0$. Hence $T_\mu(q_t)$ is strictly positive definite. \square

2.6.6 Univerality for the sequential scaling

Proposition 6. *Fix a compact K on \mathbb{R}^p , consider the sequence of kernels on K defined in (2.19), with $\sum_{l \geq 1} \lambda_l^2 < \infty$. The sequence converges uniformly on K to a universal kernel Q_∞ .*

Proof. By Proposition 5 we know that $\Lambda = \sup_{l \geq 0} \sup_{x, x' \in K} |Q_l(x, x')| < \infty$. In particular, from (2.15) we get that for each $l \geq 1$

$$\sup_{x, x' \in K} |Q_l(x, x') - Q_{l-1}(x, x')| \leq \lambda_l^2 (\sigma_b^2 + \sigma_w^2 \Lambda),$$

where we used that $|C_{l-1}/Q_{l-1}| \leq \Lambda$. From this, we conclude that the limiting kernel Q_∞ is well defined, and

$$\sup_{x, x' \in K} |Q_\infty(x, x') - Q_l(x, x')| \leq (\sigma_b^2 + \sigma_w^2 \Lambda) \sum_{l' > l} \lambda_{l'}^2,$$

which vanishes as $l \rightarrow \infty$. Moreover, it is straightforward to show that Q_∞ is a kernel, as it is the unicorn limit of kernels.

Now, with the same arguments used for proving Proposition 1, we get that for any fixed l_* , the kernel Q_{l_*} is universal. It now suffices to notice that for all $l \geq 1$, $Q_l - Q_{l-1}$ is a kernel (cf. the proof of Proposition 1), and that this fact implies that $Q_\infty - Q_{l_*}$ is a kernel, as it is the uniformly convergent sum of kernels. We can then conclude by the classical result that, given two kernels Q and R , their sum $Q + R$ is a kernel, and its RKHS \mathcal{H}_{Q+R} includes both \mathcal{H}_Q and \mathcal{H}_R (see for instance Theorem 6.24 in [Paulsen and Raghupathi, 2016](#)). \square

2.6.7 Universality on the sphere

Throughout this section, we denote as ν the standard spherical measure on \mathbb{S}^{d-1} . To prove Proposition 7, since $\sigma_b = 0$, we cannot use Proposition 2. We will hence state some preliminary results.

Lemma 14. *Let $\{A_n\}_{n \in \mathbb{N}}$ be a family of compact non-negative operators on a separable Hilbert space \mathcal{H} . Let R_n be the range of A_n and assume that $V = \text{Span}(\bigcup_{n \in \mathbb{N}} R_n)$ is dense in \mathcal{H} . Let*

$\{\alpha_n\}_{n \in \mathbb{N}}$ be a strictly positive sequence such that the sum

$$A = \sum_{n \in \mathbb{N}} \alpha_n A_n$$

converges in the operator norm. Then A is a compact strictly positive definite operator.

Proof. A is the convergent limit of a sum of compact self-adjoint operators; hence, it is compact and self-adjoint. Now, fix an arbitrary non-zero $h \in \mathcal{H}$. In order to prove that A is strictly positive, it is enough to prove that $\langle Ah, h \rangle > 0$. Denote by V_N the linear span of $\bigcup_{n \in [0:N]} R_n$. Since $V_N \subseteq V_{N+1}$ for all N , and $\bigcup_{N \in \mathbb{N}} V_N = V$ is dense in H , there exists a sequence $\{h_N\}_{N \in \mathbb{N}}$ converging to h and such that $h_N \in V_N$ for all N .

Now let us show that there must exist $n^* \in \mathbb{N}$ such that $A_{n^*} h \neq 0$. Since $\lim_{N \rightarrow \infty} \langle h, h_N \rangle = \langle h, h \rangle > 0$, there must be a N^* such that $\langle h, h_{N^*} \rangle > 0$ and so there exists $n^* \in [0 : N^*]$ and $h_{n^*} \in V_{n^*}$ such that $\langle h, h_{n^*} \rangle \neq 0$. In particular, h is not orthogonal to R_{n^*} and can not lie in the nullspace of A_{n^*} , using the fact that A_{n^*} is compact and self-adjoint and so its range and its nullspace are orthogonal (Lang, 2012). Using the spectral decomposition of non-negative compact operators, it is straightforward that $A_{n^*} h \neq 0$ implies that $\langle A_{n^*} h, h \rangle > 0$. Now, since A_n is non-negative and $\alpha_n > 0$ for all n , we have

$$\langle Ah, h \rangle = \sum_{n \in \mathbb{N}} \alpha_n \langle A_n h, h \rangle \geq \alpha_{n^*} \langle A_{n^*} h, h \rangle > 0,$$

and so we conclude. \square

Lemma 15. For all $n \in \mathbb{N}$, consider the kernel p_n on \mathbb{S}^{d-1} , defined by $p_n(x, x') = (x \cdot x')^n$, and let $T_\nu(p_n)$ be the induced integral operator on $L^2(\mathbb{S}^{d-1}, \nu)$. Denoting as R_n the range of $T_\nu(p_n)$, the subspace $V = \text{Span}(\bigcup_{n \in \mathbb{N}} R_n)$ is dense in $L^2(\mathbb{S}^{d-1}, \nu)$. Moreover, letting $V' = \text{Span}(\bigcup_{n \in \mathbb{N}} R_{2n})$ and $V'' = \text{Span}(\bigcup_{n \in \mathbb{N}} R_{2n+1})$, we have $L^2(\mathbb{S}^{d-1}, \nu) = \overline{V'} \oplus \overline{V''}$, the overline denoting the closure in $L^2(\mathbb{S}^{d-1}, \nu)$.

Proof. To prove that V is dense, first notice that for each spherical harmonic Y^{11} , we can find an operator in the form $T_\nu(P(x \cdot x'))$, for a polynomial P , which has Y in its range. Since the range of such an operator is trivially contained in V , it follows that V contains all the

¹¹We recall that a useful orthonormal basis of $L^2(\mathbb{S}^{d-1}, \nu)$ is given by the spherical harmonics (see Appendix H in Yang, 2019b for the derivation of several properties of these functions).

spherical harmonics, and so it is dense in $L^2(\mathbb{S}^{d-1}, \nu)$. Now, note that for any even n and odd n' we have

$$\int_{\mathbb{S}^{d-1}} (x \cdot z)^n (z \cdot x')^{n'} d\nu(z) = 0,$$

by an elementary symmetry argument since it is the integral on the sphere of a homogeneous polynomial of odd degree $n + n'$ in the components z_i of z . It follows that V' and V'' are orthogonal. Since their union V is dense, we conclude that $L^2(\mathbb{S}^{d-1}, \nu) = \overline{V'} \oplus \overline{V''}$. \square

Corollary 2. *With the notations of Lemma 15, assume that a sequence $\{\alpha_{n \in \mathbb{N}}\}$ is such that $A = \sum_{n \in \mathbb{N}} \alpha_n T_\nu(p_n)$ converges with respect to the operator norm on $L^2(\mathbb{S}^{d-1}, \nu)$. Then $A = A' + A''$, where $A' : \overline{V'} \rightarrow \overline{V'}$ and $A'' : \overline{V''} \rightarrow \overline{V''}$. Such a decomposition is unique and*

$$A' = \sum_{n \in \mathbb{N}} \alpha_{2n} T_\nu(p_{2n}); \quad A'' = \sum_{n \in \mathbb{N}} \alpha_{2n+1} T_\nu(p_{2n+1}),$$

both sums converging with respect to the operator norm.

Proof. It is clear that $A = A' + A''$, when both A' and A'' are defined on the whole $L^2(\mathbb{S}^{d-1}, \nu)$.

Consider any $\varphi \in L^2(\mathbb{S}^{d-1}, \nu)$. We have $A'\varphi \in \overline{V'}$, since $T_\nu(p_{2n})\varphi \in \overline{V'}$ for all n . Analogously, we can show that $A''\varphi \in \overline{V''}$. In order to conclude that we can consider the restrictions of A' and A'' to $\overline{V'}$ and $\overline{V''}$ respectively, it is enough to recall that, for compact self-adjoint operators, the nullspace is the orthogonal of the closure of the range (Lang, 2012), so that the nullspace of A' contains $\overline{V''}$ and the nullspace of A'' contains $\overline{V'}$. \square

Corollary 3. *Consider a kernel Q on the unit sphere $\mathbb{S}^{d-1} \subset \mathbb{R}^p$, such that for all $x, x' \in \mathbb{S}^{d-1}$*

$$Q(x, x') = \sum_{i \geq 0} \alpha_i (x \cdot x')^i,$$

where the α_i 's are strictly positive coefficients, and the convergence is uniform on \mathbb{S}^{d-1} . Then, $T_\nu(Q)$ is strictly positive definite.

Proof. This is a classical result (Schoenberg, 1942) that we can recover as a consequence of Lemma 14 and Corollary 3. \square

Lemma 16. *The function $f : [-1, 1] \rightarrow \mathbb{R}$, defined in (2.6), is an analytic function on $(-1, 1)$, whose expansion $f(\gamma) = \sum_{n \in \mathbb{N}} \alpha_n \gamma^n$ converges uniformly on $[-1, 1]$. Moreover, $\alpha_n > 0$ for all even $n \in \mathbb{N}$, $\alpha_1 = -1/2$ and $\alpha_n = 0$ for all odd $n \geq 3$. Moreover, the function $g : [-1, 1] \rightarrow \mathbb{R}$, defined as $g(\gamma) = f(\gamma)f'(\gamma)$, is analytic on $(-1, 1)$ and its expansion $g(\gamma) = \sum_{n \in \mathbb{N}} \beta_n \gamma^n$ has all the coefficients strictly positive and converges uniformly on $[-1, 1]$.*

Proof. The claims for f have already been proven in Lemma 9. As for g , the analyticity of f implies the one of f' , and it is easy to check the convergence on $[-1, 1]$. Moreover, all the odd Taylor coefficients of f' are strictly positive, as the even coefficients of f are. It follows that $\beta_n > 0$ for all odd n . \square

Proposition 11. *Given any non-zero $\varphi \in L^2(\mathbb{S}^{d-1}, \nu)$, there exists a $t_\varphi \in (0, 1]$ such that $\langle T_\nu(qt) \varphi, \varphi \rangle > 0$, for all $t \in (0, t_\varphi)$.*

Proof. The case $\sigma_b > 0$ has been already established in Proposition 9, hence suppose that $\sigma_b = 0$. First recall (2.24), which now reads

$$\dot{c}_t = \frac{\sigma_w^2}{2} f(c_t). \quad (2.28)$$

Deriving once more, we have

$$\ddot{c}_t = g(c_t), \quad (2.29)$$

where $g = ff'$ as in Lemma 16. Define the kernels p_n , and the subspaces V' and V'' of $L^2(\mathbb{S}^{d-1}, \nu)$, as in Lemma 15. By (2.28) and (2.29) we can write

$$c_t = c_0 + t \dot{c}_0 + \frac{t^2}{2} \ddot{c}_0 + o(t^2) = c_0 + t f(c_0) + \frac{t^2}{2} g(c_0) + o(t^2).$$

Since $\sigma_b = 0$, we have that $c_0(x, x') = x \cdot x'$, so that $c_0 = p_1$. From Lemma 16, $T_\nu(\dot{c}_0) = \sum_{n \in \mathbb{N}} \alpha_n T_\nu(p_n)$ and $T_\nu(\ddot{c}_0) = \sum_{n \in \mathbb{N}} \beta_n T_\nu(p_n)$, both sums converging in the operator norm. Moreover, $\alpha_n > 0$ for all even n and $\alpha_n = 0$ for all odd $n \geq 3$, whilst $\beta_n > 0$ for all odd n . In particular, by Corollary 2 and Lemma 14, we deduce that the restriction of $T_\nu(\dot{c}_0)|_{\overline{V'}} : \overline{V'} \rightarrow \overline{V'}$ is well defined and strictly positive, and the same holds for the restriction $T_\nu(\ddot{c}_0)|_{\overline{V''}} : \overline{V''} \rightarrow \overline{V''}$.

Now fix a non-zero $\varphi \in L^2(\mathbb{S}^{d-1}, \nu)$. By Lemma 15, we can write $\varphi = \varphi' + \varphi''$, with $\varphi' \in \overline{V'}$,

$\varphi'' \in \overline{V''}$ uniquely determined. First, suppose that $\varphi' \neq 0$. Using Corollary 1 and recalling that $c_0 = p_1$, we get

$$\langle T_\nu(c_t) \varphi, \varphi \rangle = t \langle T_\nu(\dot{c}_0)|_{\overline{V'}} \varphi', \varphi' \rangle + \langle (1 + t \alpha_1) T_\nu(p_1) \varphi'', \varphi'' \rangle + o(t) > 0,$$

for t small enough. On the other hand, for $\varphi' = 0$, we have $\varphi = \varphi''$ and so

$$\langle T_\nu(c_t) \varphi, \varphi \rangle = \langle (1 + t \alpha_1) T_\nu(p_1) \varphi'', \varphi'' \rangle + \frac{t^2}{2} \langle T_\nu(\ddot{c}_0)|_{\overline{V''}} \varphi'', \varphi'' \rangle + o(t^2) > 0$$

for t small enough. So, there is a t_φ such that, for $t \in (0, t_\varphi)$, $\langle T_\nu(c_t) \varphi, \varphi \rangle > 0$. It follows immediately that the same property is true for $T_\nu(q_t)$. \square

Lemma 17. *Let Q be a kernel on \mathbb{S}^{d-1} . Then Q is universal on \mathbb{S}^{d-1} if, and only if, $T_\nu(Q)$ is strictly positive definite on $L^2(\mathbb{S}^{d-1}, \nu)$.*

Proof. If Q is universal, $T_\nu(Q)$ is strictly positive definite by Lemma 3. On the other hand, if $T_\nu(Q)$ is strictly positive definite, it is known that its range contains all the spherical harmonics (Yang and Salman, 2019). Since the RKHS generated by Q contains the range of $T_\nu(Q)$ (see Proposition 11.17 in Paulsen and Raghupathi, 2016), it contains the linear span of the spherical harmonics, which is dense in $C(\mathbb{S}^{d-1})$ (Kounchev, 2001). Hence Q is universal. \square

Proposition 7. *Let q_t be the solution of the Cauchy problem (2.18) with $\sigma_b = 0$. Then, for all $t > 0$, q_t is universal on \mathbb{S}^{d-1} . Let Q_l be the solution of (2.19) with $\sigma_b = 0$. The sequence of these kernels converges uniformly to a universal kernel Q_∞ on \mathbb{S}^{d-1} , and for all $l \geq 2$, the kernel Q_l is universal on \mathbb{S}^{d-1} .*

Proof. For the uniform scaling, we can proceed as in the proof of Theorem 6, using Proposition 10 and Proposition 11 we can show that $T_\nu(q_t)$ is strictly positive definite on $L^2(\mathbb{S}^{d-1}, \nu)$ for all $t \in (0, 1]$. We conclude by Lemma 17 that q_t is universal on \mathbb{S}^{d-1} .

For the sequential scaling, first, we establish the universality of Q_l , for $l \geq 2$. To do so, we first notice that for $\sigma_b = 0$, the recurrence relation (2.19) for the correlation kernel reads

$$C_l(x, x') = h_l(C_{l-1}(x, x')),$$

where

$$h_l(c) = c + \frac{\lambda_l^2 \sigma_w^2}{2} \left(1 + \frac{\lambda_l^2 \sigma_w^2}{2}\right)^{-1} f(c),$$

(cf. (2.23)). Since $\left(1 + \frac{\lambda_l^2 \sigma_w^2}{2}\right)^{-1} < 1$, by Lemma 16, we see that

$$h_l(c) = \alpha_{1,l} c + \sum_{k \geq 0} \alpha_{2k,l} c^k,$$

where all the coefficients α that appear are strictly positive, and the convergence is uniform on \mathbb{S}^{d-1} . Moreover, it follows that for any $l \geq 1$, we have that

$$h_{l+1}(h_l(c)) = \sum_{k \geq 0} \beta_{k,l} c^k,$$

where all the coefficients $\beta_{k,l}$ are strictly positive, and again the convergence is uniform on \mathbb{S}^{d-1} . Since $C_2 = h_2\left(h_1\left(\frac{\sigma_w^2}{p} p_1\right)\right)$, we conclude by Corollary 3 and Lemma 17 that C_2 is universal, and so is Q_2 . Using the same arguments of the proof of Proposition 1, we conclude that for all $l \geq 2$, Q_l is ν -universal, and the sequence of these kernels converges to a ν -universal limit Q_∞ . □

2.7 Statement of authorship

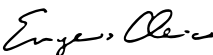
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).


Title of Paper	Stable ResNet
Publication Status	Published
Publication Details	S. Hayou, E. Clerico, B. He, G. Deligiannidis, A. Doucet, and J. Rousseau. Stable ResNet. AISTATS, 2021.

Student Confirmation

Student Name:	Eugenio Clerico
Contribution to the Paper	<p>Soufiane Hayou, Bobby He, and I equally contributed to the paper. My focus was on the theoretical side.</p> <p>My main contributions are Proposition A.1, allowing to extend the results of universality on a generic compact, along with the whole discussion about the uniform scaling and the rigorous formulation of the duality universality/expressivity, summarised by Lemma 3.</p> <p>The bulk of the experimental work was performed by Bobby He, with a strong help from Soufiane Hayou who provided, from one of his former papers, much of the Python code used. The result of stability of the gradient (Proposition 1), as well as Section 4, which gives a PAC-Bayes bound for GP regression, and the discussion of the sequential scaling are due to Soufiane Hayou.</p> <p>George Deligiannidis, Arnaud Doucet, and Judith Rousseau provided helpful insights and contributed to the writing of the paper and to the checking the proofs.</p>
Signature	
Date	24/03/2023

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:	Prof George Deligiannidis
Supervisor comments	Eugenio's description of his contributions to the paper is fair and accurate
Signature	
Date	27/03/2023

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 3

Gaussian PAC-Bayes

Wide stochastic networks: Gaussian limit and PAC-Bayesian training

Eugenio Clerico

George Deligiannidis

Arnaud Doucet

Department of Statistics, University of Oxford

CLERICO@STATS.OX.AC.UK

DELIGIAN@STATS.OX.AC.UK

DOUCET@STATS.OX.AC.UK

Editors: Shipra Agrawal and Francesco Orabona

Abstract

The limit of infinite width allows for substantial simplifications in the analytical study of overparameterised neural networks. With a suitable random initialisation, an extremely large network exhibits an approximately Gaussian behaviour. In the present work, we establish a similar result for a simple stochastic architecture whose parameters are random variables, holding both before and during training. The explicit evaluation of the output distribution allows for a PAC-Bayesian training procedure that directly optimises the generalisation bound. For a large but finite-width network, we show empirically on MNIST that this training approach can outperform standard PAC-Bayesian methods.

Keywords: Infinite width; Gaussian limit; PAC-Bayes; Stochastic networks.

1. Introduction

In recent years, overparameterised artificial neural networks with millions of nodes have shown remarkably good generalisation capabilities. This behaviour contradicts the traditional well-rooted belief that overfitting is unavoidable when the trainable parameters far outnumber the size of the training dataset. It also highlights how the complexity bounds from classical statistical learning theory (Vapnik, 2000; Bousquet et al., 2004; Shalev-Shwartz and Ben-David, 2014) are manifestly inadequate tools to assess the generalisation properties of modern neural architectures (Zhang et al., 2021). As a consequence, the last couple of decades have seen the flourishing of novel results and techniques, aiming to explain the undeniable success of overparameterised models.

A large number of trainable parameters makes the direct study of a network’s training dynamics extremely challenging. However, things become more manageable when approximations are made, as is the case in the limit of infinite width (Neal, 1995; Schoenholz et al., 2017; Yang, 2019; Hayou et al., 2019; Lee et al., 2019; Sirignano and Spiliopoulos, 2020; De Bortoli et al., 2020; Hayou et al., 2021). For a fully-connected feed-forward network, this limit consists in assuming that each layer includes an infinite number of nodes, while alternative definitions of *width* allow for extensions of this idea to encompass a vast range of architectures (Yang, 2019). Although unachievable in practice, infinitely wide networks feature the interesting property of behaving like Gaussian processes at initialisation, when all the parameters are suitably randomly initialised. This fact enable us to capture the output’s behaviour of large (but finite-size) models, both before (Matthews et al., 2018; Lee et al., 2018) and during the training (Jacot et al., 2018).

In this work, we establish a similar asymptotic result for a simple stochastic architecture, featuring a single hidden layer. For a *stochastic* network, the randomness is not limited to the initialisation but is intrinsic in the parameters, which are treated as random variables. Specifically, here we assume that each parameter follows an independent normal distribution. As the architecture’s width

approaches infinity, we show that the network’s output becomes Gaussian, with mean and covariance that can be derived from the means and standard deviations of the random parameters. We also show that under a lazy-regime assumption, where the parameters stay close to their initial values, this Gaussian behaviour is preserved throughout the training.

Part of the interest in studying stochastic networks is their role in the context of *learning with guarantees*, where the goal is to provide an upper-bound on the generalisation error without making use of any held-out test dataset. For long, in the overparameterised regime tight bounds could only be achieved under strong, and often hardly verifiable, hypotheses (Allen-Zhu et al., 2019). However, some promising non-vacuous results have been recently obtained by applying PAC-Bayesian methods to the training of stochastic classifiers (Dziugaite and Roy, 2017; Zhou et al., 2019; Pérez-Ortiz et al., 2021; Biggs and Guedj, 2022; Clerico et al., 2022).

The PAC-Bayesian theory originated from the seminal work of Shawe-Taylor and Williamson (1997), Shawe-Taylor et al. (1998), and McAllester (1998, 1999). We refer to Catoni (2007) for an extensive monograph on the topic, and to Guedj (2019) and Alquier (2021) for recent introductory overviews. It is a framework that provides upper bounds on the expected generalisation error of stochastic classifiers, with high probability over the random draw of the training dataset. The underlying idea is that if the distribution of the network’s parameters does not change much during the training, then the learnt model should not be prone to overfit.

We call PAC-Bayesian training a procedure that aims to optimise a PAC-Bayesian bound. Often this optimisation cannot be tackled directly, as the distribution of the network’s output is unknown, and one needs to sample multiple realisations of the stochastic parameters (Dziugaite and Roy, 2017; Pérez-Ortiz et al., 2021). In this paper, we propose to train a shallow wide stochastic network by exploiting the fact that it has an approximately Gaussian output. Notably, this approach allows for the direct optimisation of PAC-Bayesian bounds, even when a non-differentiable loss function, such as the 01-loss, is considered. We show empirically that the procedure that we present can bring tighter bounds and outperform standard PAC-Bayesian training methods.

As a final remark, it is worth mentioning that this is not the first work suggesting to exploit the output’s Gaussianity to train a stochastic network. For instance, Alquier et al. (2016) uses a similar approach, but limited to a linear model for binary classification. Also, Clerico et al. (2022) built on a preprint of the current paper to develop a Gaussian training method for multilayer architectures.

2. Stochastic networks

We consider a simple network $\mathbb{R}^p \rightarrow \mathbb{R}^q$, consisting of a single hidden layer made of n nodes:

$$F(x) = W^1 \phi(W^0 x), \tag{1}$$

where W^1 is a $q \times n$ matrix, W^0 a $n \times p$ matrix, and ϕ the activation function. The network is stochastic. This means that W^0 and W^1 are random variables and each time a new input is fed to the network a new realisation of them is used to evaluate the output. Concretely, we let

$$W_{ij}^1 = \frac{1}{\sqrt{n}}(\mathfrak{s}_{ij}^1 \zeta_{ij}^1 + \mathfrak{m}_{ij}^1); \quad W_{jk}^0 = \frac{1}{\sqrt{p}}(\mathfrak{s}_{jk}^0 \zeta_{jk}^0 + \mathfrak{m}_{jk}^0),$$

where $(\zeta_{ij}^1)_{i=1\dots n}^{j=1\dots q}$ and $(\zeta_{jk}^0)_{j=1\dots n}^{k=1\dots p}$ are independent families of iid standard normal random variables. We will henceforth call hyper-parameters the means \mathfrak{m} and the standard deviations \mathfrak{s} , which are deterministic quantities when conditioned on their values at initialisation (possibly random).

We are interested in the infinite-width limit of large n . We aim at showing that, as $n \rightarrow \infty$, for each fixed input x the network's output $F(x)$ converges to a multivariate normal, whose covariance matrix $Q(x) \in \mathbb{R}^q \times \mathbb{R}^q$ and mean vector $M(x) \in \mathbb{R}^q$ are deterministic functions of the hyper-parameters \mathfrak{m} and \mathfrak{s} . In short, for any fixed input x , we want to establish that

$$F(x) \rightarrow \mathcal{N}(M(x), Q(x)).^1 \quad (2)$$

Note that, for two different inputs x and x' , $F(x)$ and $F(x')$ are independent, as we assume that the stochastic parameters of the model are re-sampled every time that a new input is provided.

As a remark, by taking the limit $n \rightarrow \infty$ we mean considering a sequence of distinct networks of increasing widths, all initialised and trained in the same way. To be rigorous, one ought to add explicit superscripts (n) to the various quantities to stress their dependence on the network's width. So, one should actually write $F^{(n)}$, and say that its mean and covariance $M^{(n)}$ and $Q^{(n)}$ can be expressed in terms of $\mathfrak{m}^{(n)}$ and $\mathfrak{s}^{(n)}$. What we will show is that, for each x , $F^{(n)}(x) \rightarrow F(x) \sim \mathcal{N}(M(x), Q(x))$, where M and Q are the limits of $M^{(n)}$ and $Q^{(n)}$. However, we believe that stressing this explicit dependence on n would result in an excessively heavy notation. Therefore, we will always omit the superscript (n) , and we will freely speak of ‘‘infinite-width limit’’ of a network, with the understanding that this has to be intended as the limit of a sequence of networks.

2.1. Infinite-width limit

We start by focusing on the hidden layer, which we denote as Y^0 . Its nodes can be expressed as

$$Y_j^0(x) = \sum_{k=1}^p W_{jk}^0 x_k = \frac{1}{\sqrt{p}} \sum_{k=1}^p \mathfrak{s}_{jk}^0 \zeta_{jk}^0 x_k + \frac{1}{\sqrt{p}} \sum_{k=1}^p \mathfrak{m}_{jk}^0 x_k,$$

for any fixed input $x \in \mathbb{R}^p$. As the ζ_{jk}^0 's are iid standard Gaussian random variables, we have that

$$Y^0(x) \sim \mathcal{N}(M^0(x), Q^0(x)).$$

This means that Y^0 is a n -dimensional multivariate normal, with mean vector and covariance matrix given by

$$M_j^0(x) = \frac{1}{\sqrt{p}} \sum_{k=1}^p \mathfrak{m}_{jk}^0 x_k; \quad Q_{jj'}^0(x) = \delta_{jj'} \frac{1}{p} \sum_{k=1}^p (\mathfrak{s}_{jk}^0 x_k)^2.$$

As $Q^0(x)$ is diagonal, all the components of $Y^0(x)$ are independent, and we can actually write

$$Y_j^0(x) = \sqrt{Q_{jj}^0(x)} \bar{\zeta}_j^0 + M_j^0(x), \quad (3)$$

where the $\bar{\zeta}_j^0$'s are independent standard normals.

Now, define the random variable

$$\Phi_j^0(x) = \phi(Y_j^0(x)).$$

Clearly, we have $F_i(x) = \sum_{j=1}^n W_{ij}^1 \Phi_j^0(x)$. Expanding the components of W^1 we can write

$$F_i(x) = \frac{1}{\sqrt{n}} \sum_{j=1}^n \mathfrak{s}_{ij}^1 \zeta_{ij}^1 \Phi_j^0(x) + \frac{1}{\sqrt{n}} \sum_{j=1}^n \mathfrak{m}_{ij}^1 \Phi_j^0(x).$$

1. Clearly, to be rigorous one needs to specify which kind of convergence is intended; see Propositions 3 and 4.

For any fixed input x , in the limit $n \rightarrow \infty$, we have an infinite sum of independent random variables, which are not identically distributed. In order to establish the convergence to a multivariate normal distribution, we need to control the variance and some higher moment of these variables, and hence require that the hyper-parameters have the correct order of magnitude. This is the case when the network is suitably initialised, and the result remains true during the training, as long as the hyper-parameters stay close enough to their initial values.

Note that for any finite width n , we can explicitly evaluate the network's mean M and covariance Q . For the mean, we have

$$M_i(x) = \mathbb{E}[F_i(x)] = \frac{1}{\sqrt{n}} \sum_{j=1}^n \mathbf{m}_{ij}^1 \mathbb{E}[\Phi_j^0(x)]. \quad (4)$$

As for $Q(x)$, we have $Q_{ii'}(x) = \mathbb{C}_{ii'}[F(x)] = \mathbb{E}[F_i(x)F_{i'}(x)] - \mathbb{E}[F_i(x)]\mathbb{E}[F_{i'}(x)]$, which becomes

$$Q_{ii'}(x) = \delta_{ii'} \frac{1}{n} \sum_{j=1}^n (\mathbf{s}_{ij}^1)^2 \mathbb{E}[\Phi_j^0(x)^2] + \frac{1}{n} \sum_{j=1}^n \mathbf{m}_{ij}^1 \mathbf{m}_{i'j}^1 \mathbb{V}[\Phi_j^0(x)], \quad (5)$$

where we used the fact that the nodes of the hidden layer are independent and so the covariance of $\Phi^0(x)$ is diagonal. Once we will have established that the limit of infinite width leads to a Gaussian output, its mean and covariance will be given by the limit $n \rightarrow \infty$ of the above expressions.

We now state some rigorous results. The next proposition (see Appendix A for the proof) builds on a central limit theorem for triangular arrays, due to Bentkus (2005).

Proposition 1 *For any fixed input x and width n , define $M(x)$ and $Q(x)$ as in (4) and (5). Let $Z(x) \sim \mathcal{N}(M(x), Q(x))$ and denote as \mathcal{C} the class of measurable convex subsets of \mathbb{R}^q . Let F be defined as in (1). Then*

$$\sup_{C \in \mathcal{C}} |\mathbb{P}(F(x) \in C) - \mathbb{P}(Z(x) \in C)| \leq \kappa q^{1/4} \frac{B(\mathbf{m}, \mathbf{s})}{\sqrt{n}},$$

where $\kappa < 4$ is an absolute constant and

$$B(\mathbf{m}, \mathbf{s}) \leq q^{1/2} \frac{\frac{1}{n} \sum_{j=1}^n \sum_{i=1}^q (2|\mathbf{s}_{ij}^1|^3 + 8|\mathbf{m}_{ij}^1|^3) \mathbb{E}[|\Phi_j^0(x)|^3]}{\left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1 \dots q} (\mathbf{s}_{ij}^1)^2\right)^{3/2}}.$$

In particular, if $B(\mathbf{m}, \mathbf{s}) = o(\sqrt{n})$ for $n \rightarrow \infty$, then $F(x) - Z(x) \rightarrow 0$, in distribution.

As a corollary of the above result, if the stochastic network acts as a classifier, its performance is related to the one of its Gaussian approximation.

Corollary 2 *Assume that the network deals with a classification problem, where for each instance x there is a single correct label $y = f(x) \in \{1 \dots q\}$. With the notation of Proposition 1, for each fixed input $x \neq 0$, define as $\hat{f}(x) = \operatorname{argmax}_{i=1 \dots q} F_i(x)$ and $\bar{f}(x) = \operatorname{argmax}_{i=1 \dots q} Z_i(x)$. We have*

$$|\mathbb{P}(\hat{f}(x) = f(x)) - \mathbb{P}(\bar{f}(x) = f(x))| \leq \kappa q^{1/4} \frac{B(\mathbf{m}, \mathbf{s})}{\sqrt{n}}.$$

Proof For each $k = \{1 \dots q\}$, the set $\{z \in \mathbb{R}^q : z_k > \max_{i \neq k} z_i\}$ is convex. Hence the claim directly follows from Proposition 1. \blacksquare

2.2. Initialisation and lazy training

With a suitable random initialisation of the hyper-parameters, and in a lazy training regime, we show that, as $n \rightarrow \infty$, our stochastic network has a Gaussian limit, in the sense that the quantity B/\sqrt{n} of Proposition 1 vanishes as $n \rightarrow \infty$. For simplicity, we shall assume that the activation function $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is Lipschitz continuous (although we do not need to specify the Lipschitz constant).

We let all the network hyper-parameters be independently initialised in the following way:

$$\begin{aligned} \mathbf{m}_{jk}^0 &\sim \mathcal{N}(0, 1); & \mathbf{m}_{ij}^1 &\sim \mathcal{N}(0, 1); \\ \mathfrak{s}_{jk}^0 &= 1; & \mathfrak{s}_{ij}^1 &= 1, \end{aligned} \tag{6}$$

For convenience we write $\hat{\mathbb{P}}$ for the probability measure representing the above initialisation, while \mathbb{P} is the probability measure describing the intrinsic stochasticity of the network. These two sources of randomness are always assumed to be independent.

Proposition 3 (Initialisation) *Consider a sequence of networks of increasing width initialised according to (6), and whose activation function ϕ is Lipschitz continuous. For any fixed input $x \neq 0$, defining B as in Proposition 1, we have $\frac{B(\mathbf{m}, \mathfrak{s})}{\sqrt{n}} \rightarrow 0$, as $n \rightarrow \infty$, in probability with respect to the random initialisation $\hat{\mathbb{P}}$. More precisely, $B(\mathbf{m}, \mathfrak{s}) = O(1)$ wrt $\hat{\mathbb{P}}$, as $n \rightarrow \infty$. In particular, at the initialisation the network tends to a Gaussian limit, in distribution wrt the intrinsic stochasticity \mathbb{P} and in probability wrt $\hat{\mathbb{P}}$.*

Proof's sketch The proof is deferred to Appendix A. The main idea is that, since all the hyper-parameters are independent under (6), the standard central limit theorem yields that the upper-bound for B stated in Proposition 1 tends to a finite limit as $n \rightarrow \infty$. ■

The next proposition states that the limit will still be valid as long as the hyper-parameters do not move too much from their initialisation (lazy training).

Proposition 4 (Lazy training) *Fix a constant $J > 0$ independent of n , and assume that ϕ is Lipschitz. For a network of width n , with initial configuration $(\tilde{\mathbf{m}}, \tilde{\mathfrak{s}})$ drawn according to $\hat{\mathbb{P}}$ as in (6), denote as \mathcal{B}_J the ball*

$$\mathcal{B}_J = \{ (\mathbf{m}, \mathfrak{s}) : \|\mathbf{m}^0 - \tilde{\mathbf{m}}^0\|_{F,2}^2 + \|\mathbf{m}^1 - \tilde{\mathbf{m}}^1\|_{F,2}^2 + \|\mathfrak{s}^0 - \tilde{\mathfrak{s}}^0\|_{F,2}^2 + \|\mathfrak{s}^1 - \tilde{\mathfrak{s}}^1\|_{F,2}^2 \leq J^2 \},$$

where $\|\cdot\|_{F,2}$ denotes the 2-Frobenius norm of a matrix. Let B be defined as in Proposition 1. For any fixed input $x \neq 0$ we have $B(\mathbf{m}, \mathfrak{s}) = O(1)$ as $n \rightarrow \infty$, uniformly on \mathcal{B}_J , in probability with respect to the random initialisation $\hat{\mathbb{P}}$.

Proof's sketch The proof is rather long and technical, and is deferred to Appendix A. However, the idea is simple and consists in showing that, under the lazy training assumption $(\mathbf{m}, \mathfrak{s}) \in \mathcal{B}_J$, B undergoes a change of order $O(1)$ during the training. Since by Proposition 3 we know that B is of order $O(1)$ at the initialisation, we can conclude. ■

In the next section, we will see that the lazy training constraint can be restated in terms of a bound on the Kullback-Leibler divergence between the initial and final distributions of the stochastic parameters. This fact will allow us to ensure that the constraint is satisfied when training the network to optimise a PAC-Bayesian objective.

3. PAC-Bayesian framework

Consider a standard classification problem, where to each instance $x \in \mathcal{X} \subseteq \mathbb{R}^p$ corresponds a unique correct label $y = f(x) \in \mathcal{Y} = \{1 \dots q\}$. The goal is to build an algorithm that is able to find a good prediction of y given x . We assume that the x 's are distributed according to some probability measure \mathbb{P}_X on \mathcal{X} . To train our algorithm, we have access to a sample $S = (X_h)_{h=1 \dots m}$, which is correctly labelled (for every $X_h \in S$ we know $f(X_h)$). Each X_h is an independent draw from \mathbb{P}_X , so that $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. We let ℓ be the 01-loss:

$$\ell(\hat{y}, y) = \begin{cases} 0 & \text{if } \hat{y} = y; \\ 1 & \text{otherwise.} \end{cases}$$

We let $\hat{f}_w(x)$ denote the prediction for the instance x , for a network with parameter configuration w . The empirical loss $L_S(w) = \frac{1}{m} \sum_{x \in S} \ell(\hat{f}_w(x), f(x))$ is the average of the 01-loss on the training set, while the true loss is $L_X(w) = \mathbb{E}_X[\ell(\hat{f}_w(X), f(X))]$.

The PAC-Bayesian framework (McAllester, 1998, 1999; Catoni, 2007; Guedj, 2019; Alquier, 2021) deals with stochastic neural classifiers. We consider a prior probability measure \mathcal{P} on the random parameters, which has to be chosen independently of the specific realisation of the random dataset S used for the training. After the training, the parameters will be described by a new probability measure \mathcal{Q} (the so-called posterior), clearly S -dependent. The idea is that if \mathcal{P} and \mathcal{Q} are not too “far” from each other, then the network will generalise well.

Under the posterior, we define the expected true loss $L_X(\mathcal{Q}) = \mathbb{E}_{W \sim \mathcal{Q}}[L_X(W)]$ and the expected empirical loss $L_S(\mathcal{Q}) = \mathbb{E}_{W \sim \mathcal{Q}}[L_S(W)]$. The PAC-Bayesian bounds are upper bounds on $L_X(\mathcal{Q})$, which hold with high probability on the random draw of the training set S . They usually involve the expected empirical error $L_S(\mathcal{Q})$ and a divergence term in the form of the Kullback-Leibler divergence between \mathcal{Q} and \mathcal{P} : $\text{KL}(\mathcal{Q} \parallel \mathcal{P}) = \mathbb{E}_{\mathcal{Q}}[\log(d\mathcal{P}/d\mathcal{Q})]$. We will use the following result, due to Langford and Seeger (2001) and Maurer (2004).

Proposition 5 *Fix a data-independent prior \mathcal{P} . With probability higher than $1 - \delta$ on the choice of the training set $S = (X_h)_{h=1 \dots m}$,*

$$L_X(\mathcal{Q}) \leq \text{kl}^{-1} \left(L_S(\mathcal{Q}) \left| \frac{\text{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{m} \right. \right), \quad (7)$$

for any posterior \mathcal{Q} . Here, we have defined $\text{kl}^{-1}(u|c) = \sup\{v \in [0, 1] : \text{kl}(u \parallel v) \leq c\}$, where $\text{kl}(u \parallel v) = u \log \frac{u}{v} + (1 - u) \log \frac{1-u}{1-v}$.

We can hence devise the following training algorithm (McAllester, 1998):

- Fix $\delta \in (0, 1)$ and a prior \mathcal{P} for the network stochastic parameters;
- Collect a sample S of m iid datapoints;
- Compute the optimal posterior \mathcal{Q} minimising (7);
- Implement a stochastic network characterised by the law \mathcal{Q} .

In practice, in essentially any realistic scenario the algorithm above cannot be implemented. Hence, one has to simplify the problem requiring that \mathcal{P} and \mathcal{Q} belong to some simple class of distributions.

2. Here we assume that the training set S has size $m \geq 8$.

3.1. PAC-Bayesian training

Following the approach of [Dziugaite and Roy \(2017\)](#), we assume that both \mathcal{P} and \mathcal{Q} are multivariate normal distributions with diagonal covariance matrices. In other words, the random parameters of the network are independent normal random variables. For the posterior, \mathbf{m} and \mathbf{s} denote the N -dimensional vectors of the means and the standard deviations, while $\tilde{\mathbf{m}}$ and $\tilde{\mathbf{s}}$ refer to the prior. In short, $\mathcal{P} = \mathcal{N}(\tilde{\mathbf{m}}, \text{diag}(\tilde{\mathbf{s}}^2))$ and $\mathcal{Q} = \mathcal{N}(\mathbf{m}, \text{diag}(\mathbf{s}^2))$. In this Gaussian setting, $\text{KL}(\mathcal{Q}||\mathcal{P})$ takes a simple form:

$$\text{KL}(\mathcal{Q}||\mathcal{P}) = \frac{1}{2} \left(\sum_{\alpha} \left(\frac{\mathbf{s}_{\alpha}}{\tilde{\mathbf{s}}_{\alpha}} \right)^2 - N + \sum_{\alpha} \left(\frac{\mathbf{m}_{\alpha} - \tilde{\mathbf{m}}_{\alpha}}{\tilde{\mathbf{s}}_{\alpha}} \right)^2 + 2 \sum_{\alpha} \log \frac{\tilde{\mathbf{s}}_{\alpha}}{\mathbf{s}_{\alpha}} \right), \quad (8)$$

where the index α runs over all the components of the hyper-parameters.

Now, the most troublesome term in (7) is $L_S(\mathcal{Q})$, which in general cannot be computed explicitly. However, we can obtain a Monte Carlo (MC) estimate $\hat{L}_S(\mathcal{Q})$ of this quantity, by sampling a few realisations of the parameters from \mathcal{Q} .

Now, the idea is to perform a gradient descent (GD) optimisation on the PAC-Bayesian objective ([Dziugaite and Roy, 2017](#); [Pérez-Ortiz et al., 2021](#)). Note that (8) is differentiable with respect to \mathbf{m} and \mathbf{s} (which are the trainable hyper-parameters of the posterior). However, $\hat{L}_S(\mathcal{Q})$ has a null gradient almost everywhere, as this is the case for $L_S(w)$ for each realisation w used in the estimate. The standard way to overcome this issue is to use a surrogate of the 01-loss for the training, such as some variant of the cross-entropy ([Dziugaite and Roy, 2017](#); [Pérez-Ortiz et al., 2021](#)). Notably, although $\hat{L}_S(\mathcal{Q})$ has a null gradient, this is not the case for $L_S(\mathcal{Q})$ (see Section 4.1 and Figure 1 for more details). Hence, if we know exactly the output’s distribution of the stochastic network, we might be able to use the 01-loss directly without the need of any surrogate. This is indeed the case for the Gaussian limit, as we will see in the next section. In a similar spirit, [Alquier et al. \(2016\)](#) studied the training of a linear binary classifier with Gaussian parameters.

It is worth mentioning that similar considerations hold when using an almost everywhere constant activation function to train a stochastic network. In this regard, [Germain et al. \(2009\)](#); [Letarte et al. \(2019\)](#); [Biggs and Guedj \(2021\)](#) developed an interesting variant of PAC-Bayesian training for binary classifiers with the sign activation function ($\phi = \text{sign}$). In that setting, the simple form of the output of each layer allows for a more explicit expression of the distribution of the hidden nodes, which permits overcoming the fact that the binary activation function is non-differentiable.

4. PAC-Bayesian training in the Gaussian limit

Instead of doing the standard PAC-Bayesian training with a surrogate loss, we can train our wide stochastic network by assuming that its Gaussian approximation is exact. However, once completed the training, we will need to evaluate the final bound without such an assumption.

At the initialisation, for a network initialised according to $\hat{\mathbb{P}}$ as in (6), the Gaussian approximation is asymptotically exact for large n . Moreover, the following lemma ensures that controlling the KL divergence is enough to claim that the network is in the lazy training regime of Proposition 4. Hence, a wide stochastic network is asymptotically Gaussian throughout its PAC-Bayesian training.

Lemma 6 *Define the multivariate Gaussian distributions $\mathcal{P} = \mathcal{N}(\tilde{\mathbf{m}}, \text{diag}(\tilde{\mathbf{s}}^2)) = \mathcal{N}(\tilde{\mathbf{m}}, \text{Id})$ and $\mathcal{Q} = \mathcal{N}(\mathbf{m}, \text{diag}(\mathbf{s}^2))$ for the parameters of a stochastic network. We have*

$$\|\mathbf{m}^0 - \tilde{\mathbf{m}}^0\|_{F,2}^2 + \|\mathbf{m}^1 - \tilde{\mathbf{m}}^1\|_{F,2}^2 + \|\mathbf{s}^0 - \tilde{\mathbf{s}}^0\|_{F,2}^2 + \|\mathbf{s}^1 - \tilde{\mathbf{s}}^1\|_{F,2}^2 \leq 2\text{KL}(\mathcal{Q}||\mathcal{P}).$$

Proof From (8), we conclude noticing that $u^2 - 1 - 2 \log u \geq (u - 1)^2$, for all $u > 0$. ■

The rest of this section is organised as follows. First, we show that it is possible to get a non-zero gradient from the expected 01-loss in the Gaussian limit. Then, we discuss how to evaluate the gradients of the output’s mean and covariance with respect to the hyper-parameters. Finally, we deal with how to obtain a rigorous PAC-Bayesian bound after the training.

4.1. Training with the 01-loss

For $x \in \mathcal{X} \subseteq \mathbb{R}^p$, we want to find the correct $y = f(x)$ among q possible labels $i = 1, \dots, q$. We consider a Gaussian network with output $F(x) \sim \mathcal{N}(M(x), Q(x))$, whose random prediction is $\hat{f}(x) = \operatorname{argmax}_{i=1\dots q} F_i(x)$. Denoting as ℓ the 01-loss, it is natural to aim at minimising $\mathbb{E}[\ell(\hat{f}(x), f(x))]$ (where the expectation is over the stochastic parameters), since this quantity is actually equal to the probability of making a mistake for x : $\mathbb{P}(\hat{f}(x) \neq f(x))$. As we want to tackle the problem by performing gradient descent optimisation, if we assume that we are able to differentiate $M(x)$ and $Q(x)$ with respect to the network trainable hyper-parameters, all we need is to evaluate $\nabla_M \mathbb{E}[\ell(\hat{f}(x), y)]$ and $\nabla_Q \mathbb{E}[\ell(\hat{f}(x), y)]$.

Note that $\ell(\hat{f}(x), y)$ has a null gradient almost everywhere for any random realisation of the network. For this reason, a non-stochastic network cannot be trained directly with the 01-loss. However, this is not the case for a stochastic network. The reason for that can be intuitively explained by thinking of what happens if we try to “go down the stairs” with GD, as illustrated in Figure 1. A single realisation of the network will be a point on a horizontal step: there is no way to understand the right direction in order to go down. However, if we consider the whole stochastic distribution of the network, it spreads over all the steps, and it has a global view of the stairs. It is hence not surprising that the gradient of the expected loss is non-zero.

For binary classification tasks, the expected 01-loss reads $\mathbb{E}[\ell(\hat{f}(x), 1)] = \mathbb{P}(F_2(x) > F_1(x))$ and $\mathbb{E}[\ell(F(x), 2)] = \mathbb{P}(F_1(x) > F_2(x))$. These quantities can be computed exactly:³

$$\begin{aligned} \mathbb{E}[\ell(\hat{f}(x), 1)] &= \mathbb{P}_{\zeta \sim \mathcal{N}(0,1)} \left(\zeta > \frac{M_1(x) - M_2(x)}{\sqrt{Q_{11}(x) + Q_{22}(x) - 2Q_{12}(x)}} \right); \\ \mathbb{E}[\ell(\hat{f}(x), 2)] &= \mathbb{P}_{\zeta \sim \mathcal{N}(0,1)} \left(\zeta > \frac{M_2(x) - M_1(x)}{\sqrt{Q_{11}(x) + Q_{22}(x) - 2Q_{12}(x)}} \right). \end{aligned}$$

Clearly, the two expressions above can be written explicitly in terms of the error function erf , as ζ is distributed as a standard normal and $\mathbb{P}(\zeta > u) = \frac{1}{2}(1 - \operatorname{erf}(u/\sqrt{2}))$. It is then straightforward to see that $\mathbb{E}[\ell(\hat{f}(x), y)]$ is differentiable with respect to M and Q , with non-zero derivatives.

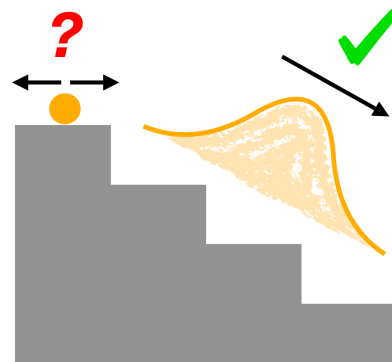


Figure 1: When “going down the stairs” via GD, each single realisation lies on a horizontal step and has an uninformative null gradient, but the whole distribution has a global view of the stairs and can find the good direction.

3. A similar result was already derived in [Alquier et al. \(2016\)](#) for a simpler linear classifier.

When there are more than two classes, things become more complicated. It is however possible to exploit the Gaussianity and obtain a MC estimator of the expected loss, whose gradient with respect to M and Q is computable and not trivially zero. We refer to Appendix B for details.

4.2. Derivatives of M and Q

We have so far established that we can effectively differentiate the expected loss with respect to M and Q . Still, to train the network we will need to evaluate the gradients with respect to the hyper-parameters \mathfrak{m} and \mathfrak{s} . Now, recall that $\Phi_k^0(x)$ is in the form $\phi(a\zeta + b)$, with $a = \sqrt{Q_{kk}^0(x)}$, $b = M_k^0(x)$, and $\zeta \sim \mathcal{N}(0, 1)$. When the activation function ϕ is simple enough, $\mathbb{E}[\phi(a\zeta + b)]$ and $\mathbb{E}[\phi(a\zeta + b)^2]$ have closed-form expressions. Exploiting this fact, it is possible to evaluate the \mathfrak{m}^0 - and \mathfrak{s}^0 -derivatives of M and Q , needed in order to train the network with gradient-based methods. This is for instance the case for $\phi = \text{ReLU}$ and $\phi = \sin$ (see Appendix C).

4.3. Final computation of the bound

Once completed the training, we need to abandon the Gaussian approximation to compute the final bound. We will follow the same approach as Dziugaite and Roy (2017) and Pérez-Ortiz et al. (2021).

Let W_1, \dots, W_N be N independent realisations of the whole set of network stochastic parameters, drawn according to \mathcal{Q} . For $\delta' \in (0, 1)$, with probability at least $1 - \delta'$ (Langford and Caruana, 2002)

$$L_S(\mathcal{Q}) \leq \text{kl}^{-1} \left(\hat{L}_S(\mathcal{Q}) \middle| \frac{1}{N} \log \frac{2}{\delta'} \right), \quad (9)$$

where kl^{-1} is defined in Proposition 5 and we have defined $\hat{L}_S(\mathcal{Q}) = \frac{1}{N} \sum_{h=1}^N L_S(W_h)$. Since kl^{-1} is increasing in its first argument, Proposition 5 yields that with probability at least $1 - \delta - \delta'$

$$L_X(\mathcal{Q}) \leq \text{kl}^{-1} \left(\text{kl}^{-1} \left(\hat{L}_S(\mathcal{Q}) \middle| \frac{1}{N} \log \frac{2}{\delta'} \right) \middle| \frac{\text{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{m} \right). \quad (10)$$

This method is often computationally very expensive, especially for large values of N . However, using a standard re-parameterisation trick from Kingma et al. (2015) helps to speed-up the evaluation, as it makes possible to obtain a realisation of the network by sampling only $d + n$ standard normals, instead of all the $p \times n^2 \times q$ stochastic parameters.

As a final remark, an alternative way to get an exact result from the Gaussian approximation is to use an upper bound, such as the one in Corollary 2, to control the finite-size correction to the expected empirical loss. However, for networks with $O(10^3)$ hidden nodes, like those that we used in our experiments, this last approach gives looser bounds compared to the method described above, at least when the number N of samples used for the MC estimate $\hat{L}_S(\mathcal{Q})$ is of order $O(10^5)$.

5. Experimental results

In this section, we present some empirical results to validate our theoretical findings. First, we compare the Gaussian predictions with the distribution of the output nodes of a wide stochastic network. Then, we report the results obtained by training a stochastic network on MNIST, and on a binary version of it, with our Gaussian method and with standard PAC-Bayesian procedures like those from Dziugaite and Roy (2017) and Pérez-Ortiz et al. (2021). On both datasets, the

Gaussian method led to tighter final generalisation bounds. The PyTorch code developed for this paper is available at <https://github.com/eclerico/WideStochNet>. For the sake of conciseness, we refer to Appendix D for an exhaustive account of the experimental details.

In order to keep the experimental setting as simple as possible, we opted for training only the means \mathbf{m} (keeping the standard deviations \mathbf{s} fixed at their initial value), similarly to what was done in [Letarte et al. \(2019\)](#). Moreover, coherently with the rest of this paper, all the networks that we used

had no bias. The PAC-Bayesian priors were chosen in a completely data-independent fashion, and coincided with the distribution of the network at initialisation, as suggested by [Dziugaite and Roy \(2017\)](#).

We start by considering a toy dataset, whose datapoints were sampled from three multivariate standard normal distributions (labelled as 1, 2, 3) in \mathbb{R}^4 , and then projected on the unit sphere in \mathbb{R}^4 . A stochastic network with one hidden layer of $n = 1200$ nodes was trained to predict from which of the three Gaussian clusters each point comes. The histograms in Figure 2 represent the distributions of the network’s output nodes, both before and after the training. They have been obtained for a single example by sampling 10^6 real-

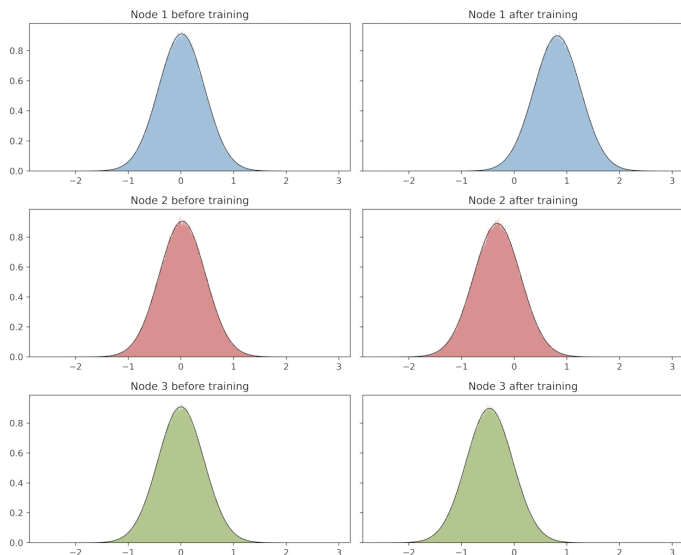


Figure 2: Distributions of the three output nodes of a wide stochastic network trained on a toy classification task. In black the theoretical predictions.

isations of the random parameters. The theoretical predictions of the Gaussian profiles are plotted in black. The agreement with the histograms is striking, showing that the network is essentially Gaussian already for $O(10^3)$ hidden nodes.

We now focus on the experiments on a binary version of the MNIST dataset, where the training dataset consisted of $m = 60000$ images. We considered a stochastic network with $n = 1200$ hidden nodes and ReLU activation function, initialised as in (6). We tried four training methods, based on different training objectives. The three “standard” PAC-Bayesian procedures used the objectives

$$\begin{aligned}
 \text{McAll} &= \bar{L}_S(\mathcal{Q}) + \sqrt{\frac{\text{KL}(\mathcal{Q}||\mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{2m}}; \\
 \text{lbd} &= \frac{\bar{L}_S(\mathcal{Q})}{(1 - \lambda/2)} + \frac{\text{KL}(\mathcal{Q}||\mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{m\lambda(1 - \lambda/2)}; \\
 \text{quad} &= \left(\sqrt{\bar{L}_S(\mathcal{Q}) + \frac{\text{KL}(\mathcal{Q}||\mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{2m}} + \sqrt{\frac{\text{KL}(\mathcal{Q}||\mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{2m}} \right)^2,
 \end{aligned} \tag{11}$$

where $\bar{L}_S(\mathcal{Q})$ is the expectation under \mathcal{Q} of the empirical cross-entropy loss divided by $\log 2$. The objective McAll is from [Dziugaite and Roy \(2017\)](#), while quad comes from [Pérez-Ortiz et al. \(2021\)](#)

and lbd was originally derived by [Thiemann et al. \(2017\)](#) and later used by [Pérez-Ortiz et al. \(2021\)](#). In lbd, $\lambda \in (0, 1)$ is also a trainable parameter.

As we are dealing with binary classification, for the ‘‘Gaussian’’ method (described Section 4), the expected value $L_S(\mathcal{Q})$ of the 01-loss can be evaluated directly (see Section 4.1). We could hence directly optimise (7), using the objective

$$\text{invkl} = \text{kl}^{-1} \left(L_S(\mathcal{Q}) \left| \frac{\text{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{m} \right. \right). \quad (12)$$

Table 1 illustrates the results of the experiment. The column ‘‘Bound’’ reports the values of the PAC-Bayesian bound (10). For the upper bound (9) on the empirical error, we used $N = 150000$ independent realisations of the net, $\delta' = 0.01$, and $\delta = 0.025$, so that the final generalisation bounds hold with probability higher than 0.965 on the random selection of the training set. The column ‘‘Test Error’’ reports the average test error on a held-out dataset and its standard deviation. These values were evaluated on 10000 independent realisations of the test error. The two next columns refer to quantities computed within the Gaussian approximation: ‘‘G Bound’’ is the bound given by (7) and ‘‘G Loss’’ is the expected 01-loss. ‘‘Penalty’’ is the quantity $(\text{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta})/m$.

The ‘‘Gaussian’’ method yielded a tighter final bound than the ‘‘standard’’ ones. Yet, the best test error is achieved by McAll. It is worth noting that the final bound obtained with

McAll is slightly worse than the one from [Dziugaite and Roy \(2017\)](#), where for a similar network of 1200 hidden nodes a final bound of .179 was obtained, whilst our result is .1921. However, our setting is simpler: our network has no bias, the standard deviations are not trained, and there is no choice of the optimal prior among different initialisations.

Finally, we report the results of a similar experiment on the full MNIST dataset (with the original 10 labels). The network is essentially the same one used for binary MNIST, with 1200 hidden nodes and ReLU activation function. The main difference is that now we have 10 output nodes. For the ‘‘standard’’ methods, we trained on the same objectives (11) as before, although this time we used a bounded version of the cross-entropy loss, as in [Pérez-Ortiz et al. \(2021\)](#). $\bar{L}_S(\mathcal{Q})$ is the expected value under \mathcal{Q} of this bounded cross-entropy, averaged on the training set. The ‘‘Gaussian’’ method used the objective (12), where $L_S(\mathcal{Q})$ is again the expected empirical 01-loss. Actually, as we were dealing with more than two classes, we could not exactly compute the expected 01-loss, since we do not have a simple closed-form expression for it, and we proceeded as described in Appendix B.

Table 2 reports the results of the experiment on the full MNIST dataset, where for the estimate of the final bounds we again used $N = 150000$, $\delta' = 0.01$, and $\delta = 0.025$. Once more, the

Table 1: Binary MNIST

Method	Bound	Test error	G Bound	G Loss	Penalty
invkl	.1773	.0694 \pm .0040	.1741	.0676	.0492
McAll	.1978	.0456 \pm .0025	.1947	.0428	.1006
lbd	.1856	.0543 \pm .0030	.1825	.0520	.0752
quad	.1855	.0533 \pm .0030	.1823	.0515	.0757

Table 2: MNIST

Method	Bound	Test Error	G Bound	G Loss	Penalty
invkl	.2807	.1083 \pm .0039	.2773	.1114	.0821
McAll	.4158	.3189 \pm .0097	.4120	.3265	.0155
lbd	.3736	.2639 \pm .0085	.3699	.2717	.0216
quad	.3735	.2637 \pm .0083	.3698	.2716	.0217

“Gaussian” method obtained a tighter result, with almost a 0.1-gap with the bounds achieved by the other procedures. This time, the “Gaussian” method also attained the tightest test error. It is worth noticing that the PAC-Bayesian penalties of the standard methods are much lower than the respective losses⁴, something that did not occur in Table 1. We conjecture that this behaviour is due to the different rescaling of the cross-entropy loss. On the other hand, this is not the case for the Gaussian method, as the loss does not require any rescaling.

6. Conclusions and perspectives

In the present work, we derive a Gaussian limit for a simple one-layer stochastic architecture, and point out how this result can be used in practice for the PAC-Bayesian training of wide shallow networks. First, we rigorously prove the validity of the limit at the initialisation and in a lazy training regime. Then, we show empirically that the proposed training method can outperform some standard PAC-Bayesian training procedures.

A main limitation of our approach is that it is limited to shallow networks with a single hidden layer. Indeed, our approach to establish the Gaussian limit relies on the fact that the hidden nodes are independent. This is not true anymore for any subsequent layer, and hence the CLT result that we use is no longer applicable. It is however worth mentioning that all the covariance matrices of the hidden layers are almost diagonal at the initialisation (as it is easy to check that the non-diagonal elements scale as $1/\sqrt{n}$) and a lazy-training constraint equivalent to the one in Proposition 4 might be enough to help establishing a rigorous Gaussian limit holding for multilayer architectures. In any case, even if one were able to use a limit theorem holding for the sum of weakly dependent nodes, evaluating the output’s law of the network would require the knowledge of the (non-diagonal) covariance matrices of the hidden layers.⁵ As we are looking at wide networks, the storage of these matrices would require a considerable amount of computational memory. Nevertheless, it is still possible to exploit our Gaussian PAC-Bayesian training ideas for multilayer architectures. This was recently done by Clerico et al. (2022), which built on our work to obtain PAC-Bayesian bounds using the fact that the network’s output is Gaussian when conditioned on the hidden layers.

As a final remark, in the present work we did not treat the case of a network with biases. This is likely to be an elementary extension, which should not require much additional work.

Acknowledgments

EC is partially supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the grant EP/R513295/1 (DTP scheme) and AD by EPSRC CoSInES EP/R034710/1. AD acknowledges support of the UK Defence Science and Technology Laboratory (DSTL) and EPSRC grant EP/R013616/1. This is part of the collaboration between US DOD, UK MOD and UK EPSRC under the Multidisciplinary University Research Initiative. The authors would like to thank Jian Qian for the valuable comments and suggestions.

4. Training with longer time did not bring any relevant improvement, as the GD descent appeared to have already stabilised.

5. Although the non diagonal elements are expected to scale as $1/\sqrt{n}$, the fact that they usually appear in sums of $O(n)$ terms can make their contribution non negligible. This was confirmed by a few empirical tests were we tried to only consider the diagonal elements of the covariance matrices of the central layers, and obtained inconsistencies between the predicted and the empirical output laws.

References

- Z. Allen-Zhu, Y. Li, and Y. Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *NeurIPS*, 2019.
- P. Alquier. User-friendly introduction to PAC-Bayes bounds. *arXiv:2110.11216*, 2021.
- P. Alquier, J. Ridgway, and N. Chopin. On the properties of variational approximations of Gibbs posteriors. *Journal of Machine Learning Research*, 17, 2016.
- V. Bentkus. A Lyapunov-type bound in \mathbb{R}^d . *Theory of Probability & Its Applications*, 49(2), 2005.
- F. Biggs and B. Guedj. Differentiable PAC-Bayes objectives with partially aggregated neural networks. *Entropy*, 23(10), 2021.
- F. Biggs and B. Guedj. Non-vacuous generalisation bounds for shallow neural networks. *ICML*, 2022.
- O. Bousquet, S. Boucheron, and G. Lugosi. *Introduction to Statistical Learning Theory*. Springer, 2004.
- O. Catoni. PAC-Bayesian supervised classification: The thermodynamics of statistical learning. *IMS Lecture Notes Monograph Series*, 2007.
- E. Clerico, G. Deligiannidis, and A. Doucet. Conditionally Gaussian PAC-Bayes. *AISTATS*, 2022.
- V. De Bortoli, A. Durmus, X. Fontaine, and U. Simsekli. Quantitative propagation of chaos for SGD in wide neural networks. *NeurIPS*, 2020.
- G. K. Dziugaite and D. M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *UAI*, 2017.
- P. Germain, A. Lacasse, F. Laviolette, and M. Marchand. PAC-Bayesian learning of linear classifiers. *ICML*, 2009.
- B. Guedj. A primer on PAC-Bayesian learning. *Proceedings of the Second congress of the French Mathematical Society*, 2019.
- S. Hayou, A. Doucet, and J. Rousseau. On the impact of the activation function on deep neural networks training. *ICML*, 2019.
- S. Hayou, E. Clerico, B. He, G. Deligiannidis, A. Doucet, and J. Rousseau. Stable resnet. *AISTATS*, 2021.
- A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *NeurIPS*, 2018.
- D.P. Kingma, T. Salimans, and M. Welling. Variational dropout and the local reparameterization trick. *NeurIPS*, 2015.
- J. Langford and R. Caruana. (Not) bounding the true error. *NeurIPS*, 2002.

- J. Langford and M. Seeger. Bounds for averaging classifiers. *CMU tech report*, 2001.
- J. Lee, Y. Bahri, R. Novak, S. Schoenholz, J. Pennington, and J. Sohl-Dickstein. Deep neural networks as Gaussian processes. *ICLR*, 2018.
- J. Lee, L. Xiao, S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. *NeurIPS*, 2019.
- G. Letarte, P. Germain, B. Guedj, and F. Laviolette. Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks. *NeurIPS*, 2019.
- A. G. Matthews, J. Hron, M. Rowland, R.E. Turner, and Z. Ghahramani. Gaussian process behaviour in wide deep neural networks. *ICLR*, 2018.
- A. Maurer. A note on the PAC Bayesian theorem. *arXiv:0411099*, 2004.
- D.A. McAllester. Some PAC-Bayesian theorems. *COLT*, 1998.
- D.A. McAllester. PAC-Bayesian model averaging. *COLT*, 1999.
- R. M. Neal. Bayesian learning for neural networks. *Springer Science & Business Media*, 118, 1995.
- M. Pérez-Ortiz, O. Risvaplata, J. Shawe-Taylor, and C. Szepesvári. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22, 2021.
- S. Schoenholz, J. Gilmer, S. Ganguli, and J. Sohl-Dickstein. Deep information propagation. *ICLR*, 2017.
- S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.
- J. Shawe-Taylor and R.C. Williamson. A PAC analysis of a Bayesian estimator. *COLT*, 1997.
- J. Shawe-Taylor, P.L. Bartlett, R.C. Williamson, and M. Anthony. Structural risk minimization over data-dependent hierarchies. *IEEE Transactions on Information Theory*, 44(5), 1998.
- J. Sirignano and K. Spiliopoulos. Mean field analysis of neural networks: A law of large numbers. *SIAM Journal on Applied Mathematics*, 80(2), 2020.
- N. Thiemann, C. Igel, O. Wintenberger, and Y. Seldin. A strongly quasiconvex PAC-Bayesian bound. *ALT*, 2017.
- V.N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.
- G. Yang. Tensor programs I: Wide feedforward or recurrent neural networks of any architecture are gaussian processes. *NeurIPS*, 2019.
- C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3), 2021.
- W. Zhou, V. Veitch, M. Austern, R.P. Adams, and P. Orbanz. Non-vacuous generalization bounds at the imagenet scale: a PAC-Bayesian compression approach. *ICLR*, 2019.

Appendix A. Omitted proofs

Throughout this section we use several notations for the norms of vectors and matrices. For $\gamma \geq 1$ and a vector v , $\|v\|_\gamma = (\sum_i |v_i|^\gamma)^{1/\gamma}$. If A is a matrix, we define $\|A\|_{F,\gamma} = (\sum_{ij} |A_{ij}|^\gamma)^{1/\gamma}$ and $\|A\|_\gamma = \sup_{v:\|v\|_\gamma=1} \|Av\|_\gamma$. We also recall that \mathbb{P} denotes the intrinsic stochasticity of the network, while $\hat{\mathbb{P}}$ is the randomness due to the initialisation. These two sources of stochasticity are always supposed to be mutually independent. We denote as \mathbb{E} the expectation wrt \mathbb{P} , and as $\hat{\mathbb{E}}$ the one wrt $\hat{\mathbb{P}}$. Moreover we write $\Gamma = O_{\hat{\mathbb{P}}}(n^\gamma)$ to mean that $\limsup_{n \rightarrow \infty} \frac{|\Gamma|}{n^\gamma} < \infty$ in probability wrt $\hat{\mathbb{P}}$, and $\Gamma = \Omega_{\hat{\mathbb{P}}}(n^\gamma)$ for $\limsup_{n \rightarrow \infty} \frac{|\Gamma|}{n^\gamma} > 0$ in probability wrt $\hat{\mathbb{P}}$.

We want to prove a rigorous result of convergence to the Gaussian limit of wide stochastic networks. We will essentially make use of the next result, due to [Bentkus \(2005\)](#).

Theorem 7 *Let X_1, \dots, X_n be independent random vectors in \mathbb{R}^q , such that $\mathbb{E}[X_j] = 0$ for all j . Let $Y = \frac{1}{\sqrt{n}} \sum_{j=1}^n X_j$ and assume that the covariance matrix $\mathbb{C}[Y]$ is non singular. Let $Z \sim \mathcal{N}(0, \mathbb{C}[Y])$. Denote as $\frac{1}{\sqrt{\mathbb{C}[Y]}}$ the inverse of the positive square root of the matrix $\mathbb{C}[Y]$, and let $B_j = \mathbb{E}[\|\frac{1}{\sqrt{\mathbb{C}[Y]}} X_j\|_2^3]$ and $B = \frac{1}{n} \sum_{j=1}^n B_j$. Let \mathcal{C} denote the class of all convex subsets of \mathbb{R}^p . Then, there exists an absolute positive constant $\kappa < 4$ such that*

$$\sup_{C \in \mathcal{C}} |\mathbb{P}(Y \in C) - \mathbb{P}(Z \in C)| \leq \kappa q^{1/4} \frac{B}{\sqrt{n}}. \quad (13)$$

Our goal is to prove a Gaussian limit as $n \rightarrow \infty$ for $F(x)$, whose components are given by

$$F_i(x) = \frac{1}{\sqrt{n}} \sum_{j=1}^n \mathfrak{s}_{ij}^1 \zeta_{ij}^1 \Phi_j^0(x) + \frac{1}{\sqrt{n}} \sum_{j=1}^n \mathfrak{m}_{ij}^1 \Phi_j^0(x).$$

Let us denote by X_j the q -dimensional vector $X_j = (X_{1j} \dots X_{qj})$, with

$$X_{ij} = \mathfrak{s}_{ij}^1 \zeta_{ij}^1 \Phi_j^0(x) + \mathfrak{m}_{ij}^1 (\Phi_j^0(x) - \mathbb{E}[\Phi_j^0(x)]).$$

Since all the ζ_{ij}^1 's and the Φ_j^0 's are independent, the X_j 's constitute a family of n centred independent q -dimensional random vectors (wrt the intrinsic network stochasticity \mathbb{P}).

Clearly, we have $F(x) = \mathbb{E}[F(x)] + \frac{1}{\sqrt{n}} \sum_{j=1}^n X_j$. Let us define $Y = \frac{1}{\sqrt{n}} \sum_{j=1}^n X_j$. Note that, for all x , the covariance matrix $\mathbb{C}[Y]$ is given by $Q(x)$ (defined in (5)), no matter if Y is Gaussian or not. Using the same notations of Theorem 7, assuming that $\mathbb{C}[Y]$ is non-singular, we have

$$\sup_{C \in \mathcal{C}} |\mathbb{P}(Y \in C) - \mathbb{P}(Z \in C)| \leq \kappa q^{1/4} \frac{B}{\sqrt{n}}.$$

To prove that the Y behaves as a Gaussian for large n , we will show that $B = O(1)$ for large n . We can easily upperbound each B_j as

$$B_j \leq \frac{1}{\lambda[Y]^{3/2}} \mathbb{E}[\|X_j\|_2^3],$$

where $\lambda[Y] > 0$ is the smallest eigenvalue of $\mathbb{C}[Y]$. For simplicity, we have omitted the dependence of these quantities on \mathfrak{m} and \mathfrak{s} . We will often do so throughout this section, in order to lighten the notation.

Define $G = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|X_j\|_2^3]$ and $\Lambda = \frac{1}{n} \sum_{j=1}^n \lambda[X_j]$. Clearly, as the X_j 's are independent, we have $\mathbb{C}[Y] = \frac{1}{n} \sum_{j=1}^n \mathbb{C}[X_j]$. In particular, we can easily find that $\lambda[Y] \geq \Lambda$.

We summarise what we have so far in the next lemma.

Lemma 8 *With the notations introduced above, assuming that $\mathbb{C}[Y]$ is not singular, we have*

$$B \leq \frac{1}{n} \sum_{j=1}^n \frac{1}{\Lambda^{3/2}} \mathbb{E}[\|X_j\|_2^3] = \frac{G}{\Lambda^{3/2}}.$$

Now, from Hölder's inequality we have $\|X_j\|_2 \leq \|\mathbf{1}_q\|_6 \|X_j\|_3 = q^{1/6} \|X_j\|_3$, and so

$$\|X_j\|_2^3 \leq q^{1/2} \|X_j\|_3^3 = q^{1/2} \sum_{i=1}^q |X_{ij}|^3. \quad (14)$$

Then, with some simple algebraic manipulation, and applying Jensen's inequality, we obtain

$$\mathbb{E}[|X_{ij}|^3] \leq (|\mathfrak{s}_{ij}^1|^3 \mathbb{E}[|\zeta_{ij}^1|^3] + 8|\mathfrak{m}_{ij}^1|^3) \mathbb{E}[|\Phi_j^0(x)|^3].$$

For convenience, we introduce the following notations

$$H_{ij} = (2|\mathfrak{s}_{ij}^1|^3 + 8|\mathfrak{m}_{ij}^1|^3) \mathbb{E}[|\Phi_j^0(x)|^3]; \quad H_j = \sum_{i=1}^q H_{ij}; \quad H = \frac{1}{n} \sum_{j=1}^n H_j,$$

so that we have $G \leq q^{1/2} H$, since $\mathbb{E}[|\zeta|^3] = 2\sqrt{2/\pi} < 2$ for $\zeta \sim \mathcal{N}(0, 1)$.

On the other hand, we can find a lowerbound for Λ as well. Indeed, first we can notice that

$$\mathbb{C}_{ii'}[X_j] = \delta_{ii'} (\mathfrak{s}_{ij}^1)^2 \mathbb{E}[\Phi_j^0(x)^2] + \mathfrak{m}_{ij}^1 \mathfrak{m}_{i'j}^1 \mathbb{V}[\Phi_j^0(x)].$$

The first term is a diagonal matrix, while the second one is non-negative definite. Hence we can write

$$\lambda[X_j] \geq \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1 \dots q} (\mathfrak{s}_{ij}^1)^2.$$

Defining

$$\Theta = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1 \dots q} (\mathfrak{s}_{ij}^1)^2 \leq \Lambda \quad (15)$$

we have the following corollary of Lemma 8.

Corollary 9 *With the same notations as above, if $\mathbb{C}[Y]$ is non singular, we have*

$$B \leq q^{1/2} \frac{H}{\Theta^{3/2}}.$$

Note that both H and Θ can be evaluated explicitly, given the parameters of the networks, as long as ϕ allows for an explicit evaluation of $\mathbb{E}[|\Phi^0(x)|^\gamma]$, for $\gamma = 1, 2, 3$. This means that we can give an exact upper bound to the finite-size error of the predicted 01-loss, for any configuration of the network.

We can now prove Proposition 1 and Corollary 2 from the main text.

Proposition 1 For any fixed input x and width n , define $M(x)$ and $Q(x)$ as in (4) and (5). Let $Z(x) \sim \mathcal{N}(M(x), Q(x))$ and denote as \mathcal{C} the class of measurable convex subsets of \mathbb{R}^q . Let F be defined as in (1). Then

$$\sup_{C \in \mathcal{C}} |\mathbb{P}(F(x) \in C) - \mathbb{P}(Z(x) \in C)| \leq \kappa q^{1/4} \frac{B(\mathbf{m}, \mathfrak{s})}{\sqrt{n}},$$

where $\kappa < 4$ is an absolute constant and

$$B(\mathbf{m}, \mathfrak{s}) \leq q^{1/2} \frac{\frac{1}{n} \sum_{j=1}^n \sum_{i=1}^q (2|\mathfrak{s}_{ij}^1|^3 + 8|\mathbf{m}_{ij}^1|^3) \mathbb{E}[|\Phi_j^0(x)|^3]}{\left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1 \dots q} (\mathfrak{s}_{ij}^1)^2\right)^{3/2}}.$$

In particular, if $B(\mathbf{m}, \mathfrak{s}) = o(\sqrt{n})$ for $n \rightarrow \infty$, then $F(x) - Z(x) \rightarrow 0$, in distribution.

Proof The result is a straight consequence of Theorem 7 and Corollary 9. Note that $\mathbb{C}[Y]$ is non singular as long as all the components of \mathfrak{s} are non-zero, so as long as the bound in the statement is finite. \blacksquare

In the next section we show how, with a suitable random initialisation, we can assure that the network has an almost Gaussian behaviour. Successively, we will show that this behaviour is preserved during training, as long as the hyper-parameters do not move too much from their initial values.

A.1. Initialisation

We consider the random initialisation:

$$\begin{aligned} \mathbf{m}_{jk}^0 &\sim \mathcal{N}(0, 1); & \mathbf{m}_{ij}^1 &\sim \mathcal{N}(0, 1); \\ \mathfrak{s}_{jk}^0 &= 1; & \mathfrak{s}_{ij}^1 &= 1, \end{aligned} \tag{6}$$

As now we have two sources of randomness (the initialisation and the intrinsic stochasticity of the network) to avoid confusion we will denote as $\hat{\mathbb{E}}, \hat{\mathbb{P}}, \hat{\mathbb{V}}$ the expectations, probabilities and variances with respect to the initialisation, whilst \mathbb{E}, \mathbb{P} and \mathbb{V} refer to the network intrinsic stochasticity.

Lemma 10 Define H and Θ as in the previous section for a network with parameters $(\mathbf{m}, \mathfrak{s})$ distributed according to $\hat{\mathbb{P}}$, as in (6). Assume that ϕ is Lipschitz continuous. Then, for any fixed $x \neq 0$, $H \rightarrow h > 0$ and $\Theta \rightarrow \theta > 0$ in probability as $n \rightarrow \infty$, with respect to the random initialisation, where both h and θ are finite.

Proof First notice that, fixed an input $x \neq 0$ and fixed n , all the Φ_j^0 's are iid, with respect to $\hat{\mathbb{P}}$, as all the components of \mathbf{m}^0 and the \mathfrak{s}_0 are. As a consequence all the H_j 's are iid with respect to $\hat{\mathbb{P}}$ (note that they have different distribution for different n as the law of the Φ_j^0 's depends on n). Now, thanks to the fact that ϕ is Lipschitz continuous, we have that $\limsup_{n \rightarrow \infty} \hat{\mathbb{V}}[H_j] < \infty$. Hence, by a standard application of the CLT for triangular arrays, we get that

$$H - \hat{\mathbb{E}}[H] = \frac{1}{n} \sum_{j=1}^n (H_j - \hat{\mathbb{E}}[H_j]) \rightarrow 0$$

in distribution, and hence in probability, as 0 is a constant. It is quickly verified that the limit $h = \lim_{n \rightarrow \infty} \hat{\mathbb{E}}[H]$ exists, finite and positive. The proof for Θ is analogous. ■

Now we can easily prove Proposition 3.

Proposition 3 *Consider a sequence of networks of increasing width initialised according to (6), and whose activation function ϕ is Lipschitz continuous. For any fixed input $x \neq 0$, defining B as in Proposition 1, we have $\frac{B(\mathbf{m}, \mathbf{s})}{\sqrt{n}} \rightarrow 0$, as $n \rightarrow \infty$, in probability with respect to the random initialisation $\hat{\mathbb{P}}$. More precisely, $B(\mathbf{m}, \mathbf{s}) = O(1)$ wrt $\hat{\mathbb{P}}$, as $n \rightarrow \infty$. In particular, at the initialisation the network tends to a Gaussian limit, in distribution wrt the intrinsic stochasticity \mathbb{P} and in probability wrt $\hat{\mathbb{P}}$.*

Proof It is a straight consequence of Lemma 10. ■

A.2. Lazy training

We have established that the Gaussian limit holds at initialisation. In the present section we will see that, as far as the hyper-parameters of the network do not move too much from their initial values, the limit keeps its validity.

Proposition 4 *Fix a constant $J > 0$ independent of n , and assume that ϕ is Lipschitz. For a network of width n , with initial configuration $(\tilde{\mathbf{m}}, \tilde{\mathbf{s}})$ drawn according to $\hat{\mathbb{P}}$ as in (6), denote as \mathcal{B}_J the ball*

$$\mathcal{B}_J = \{(\mathbf{m}, \mathbf{s}) : \|\mathbf{m}^0 - \tilde{\mathbf{m}}^0\|_{F,2}^2 + \|\mathbf{m}^1 - \tilde{\mathbf{m}}^1\|_{F,2}^2 + \|\mathbf{s}^0 - \tilde{\mathbf{s}}^0\|_{F,2}^2 + \|\mathbf{s}^1 - \tilde{\mathbf{s}}^1\|_{F,2}^2 \leq J^2\},$$

where $\|\cdot\|_{F,2}$ denotes the 2-Frobenius norm of a matrix. Let B be defined as in Proposition 1. For any fixed input $x \neq 0$ we have $B(\mathbf{m}, \mathbf{s}) = O(1)$ as $n \rightarrow \infty$, uniformly on \mathcal{B}_J , in probability with respect to the random initialisation $\hat{\mathbb{P}}$.

Proof For convenience we will write with a tilde all the quantities relative to the network at initialisation. We denote with a Δ the difference between the final and the initial values of these quantities. For instance, $\Theta = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1 \dots q} (\mathbf{s}_{ij}^1)^2$, $\tilde{\Theta} = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\tilde{\Phi}_j^0(x)^2] \min_{i=1 \dots q} (\tilde{\mathbf{s}}_{ij}^1)^2$, and $\Delta\Theta = \Theta - \tilde{\Theta}$.

We will show that for $n \rightarrow \infty$, $\Theta = \Omega_{\hat{\mathbb{P}}}(1)$ and $G = O_{\hat{\mathbb{P}}}(1)$ uniformly on \mathcal{B}_J , so that we can conclude using that $\Lambda \geq \Theta$ and Lemma 8.

Fix an input x . First, we need a bound on $\|\Delta\Phi^0(x)\|_2 = \|\Phi^0(x) - \tilde{\Phi}^0(x)\|_2$. We have that $\Phi^0(x) = \phi(Y^0(x))$. Hence, letting L be the Lipschitz constant of ϕ , we have $\|\Delta\Phi^0(x)\|_2 \leq L\|\Delta Y^0(x)\|_2$. Now, as $\Delta Y_j^0(x) = \frac{1}{\sqrt{p}} \sum_{k=1}^p \Delta \mathbf{m}_{jk}^0 x_k + \frac{1}{\sqrt{p}} \sum_{k=1}^p \Delta \mathbf{s}_{jk}^0 \zeta_{jk}^0 x_k$, we have

$$\|\Delta\Phi^0(x)\|_2 \leq \frac{L}{\sqrt{p}} (\|\Delta \mathbf{m}^0\|_2 + \|\Delta \mathbf{s}^0 \odot \zeta^0\|_2) \|x\|_2,$$

where \odot denotes the Hadamard product.

Notice that we have

$$\mathbb{E}[\|\Delta \mathbf{s}^0 \odot \zeta^0\|_2^2] \leq \mathbb{E}[\|\Delta \mathbf{s}^0 \odot \zeta^0\|_{F,2}^2] = \sum_{j=1}^n \sum_{k=1}^p (\Delta \mathbf{s}_{jk}^0)^2 \mathbb{E}[(\zeta_{jk}^0)^2] = \|\Delta \mathbf{s}^0\|_{F,2}^2 \leq J^2$$

uniformly in \mathcal{B}_J , where as usual the expectation \mathbb{E} is the one with respect to the intrinsic stochasticity of the network, due to the ζ 's. We can define a constant $C \geq 0$, independent of n , such that

$$\mathbb{E}[\|\Delta\Phi^0(x)\|_2^2] \leq \frac{4L^2J^2\|x\|_2^2}{p} = C^2$$

uniformly in \mathcal{B}_J , as $\|\Delta\mathbf{m}^0\|_2 \leq \|\Delta\mathbf{m}^0\|_{F,2} \leq J$.

Now, recalling the definition of Θ and using that $\mathbf{s}^1 = 1 + \Delta\mathbf{s}^1$, we have

$$\Theta = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1\dots q} (\mathbf{s}_{ij}^1)^2 \geq \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] (1 - 2 \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1|).$$

We will show that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \rightarrow \tilde{\Theta}$ and $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| \rightarrow 0$.

First notice that

$$\left| \frac{1}{n} \mathbb{E}[\|\Phi^0(x)\|_2^2] - \frac{1}{n} \mathbb{E}[\|\tilde{\Phi}^0(x)\|_2^2] \right| \leq \frac{2}{n} \mathbb{E}[|\tilde{\Phi}^0(x) \cdot \Delta\Phi^0(x)|] + \frac{1}{n} \mathbb{E}[\|\Delta\Phi_j^0(x)\|_2^2].$$

We know that $\frac{1}{n} \mathbb{E}[\|\tilde{\Phi}^0(x)\|_2^2] = \tilde{\Theta}$ by definition. On the other hand we have

$$\begin{aligned} \frac{2}{n} \mathbb{E}[|\tilde{\Phi}^0(x) \cdot \Delta\Phi^0(x)|] &\leq \frac{2}{n} \mathbb{E}[\|\tilde{\Phi}^0(x)\|_2 \|\Delta\Phi^0(x)\|_2] \\ &\leq \frac{2C}{\sqrt{n}} \left(\frac{1}{n} \mathbb{E}[\|\tilde{\Phi}^0(x)\|_2^2] \right)^{1/2} = \frac{2C\tilde{\Theta}^{1/2}}{\sqrt{n}} = O_{\mathbb{P}}(1/\sqrt{n}). \end{aligned}$$

Since $\frac{1}{n} \mathbb{E}[\|\Delta\Phi_j^0(x)\|_2^2] \leq C^2/n$, we have that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] - \tilde{\Theta} \rightarrow 0$ uniformly in \mathcal{B}_J , in probability with respect to the random initialisation $\hat{\mathbb{P}}$.

We still need to show that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| \rightarrow 0$. Again we can decompose the term in Φ^0 and we have

$$\begin{aligned} \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| \\ = \frac{1}{n} \sum_{j=1}^n (\mathbb{E}[\tilde{\Phi}_j^0(x)^2] + 2\mathbb{E}[\tilde{\Phi}_j^0(x)\Delta\Phi_j^0(x)] + \mathbb{E}[\Delta\Phi_j^0(x)^2]) \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1|. \end{aligned}$$

Clearly, for every j we have $\min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| \leq J$, and so we can write

$$\begin{aligned} \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\Phi_j^0(x)^2] \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| &\leq \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\tilde{\Phi}_j^0(x)^2] \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| \\ &\quad + \frac{2J}{n} (\mathbb{E}[|\tilde{\Phi}^0(x) \cdot \Delta\Phi^0(x)|] + \mathbb{E}[\|\Delta\Phi^0(x)\|_2^2]) \end{aligned}$$

uniformly in \mathcal{B}_J . We know already that $\frac{1}{n} (2\mathbb{E}[|\tilde{\Phi}^0(x) \cdot \Delta\Phi^0(x)|] + \mathbb{E}[\|\Delta\Phi^0(x)\|_2^2]) = O_{\mathbb{P}}(1/\sqrt{n})$. As for the other term, we have

$$\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\tilde{\Phi}_j^0(x)^2] \min_{i=1\dots q} |\Delta\mathbf{s}_{ij}^1| \leq \frac{1}{\sqrt{n}} \left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\tilde{\Phi}_j^0(x)^4] \right)^{1/2} \left(\sum_{j=1}^n \min_{i=1\dots q} (\Delta\mathbf{s}_{ij}^1)^2 \right)^{1/2}.$$

Using an argument analogous to that in the proof of Proposition 3, we have that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\tilde{\Phi}_j^0(x)^4]$ has a finite limit (in probability wrt $\hat{\mathbb{P}}$). On the other hand, we have

$$\sum_{j=1}^n \min_{i=1\dots q} (\Delta \mathfrak{s}_{ij}^1)^2 \leq \sum_{j=1}^n \sum_{i=1}^q (\Delta \mathfrak{s}_{ij}^1)^2 = \|\Delta \mathfrak{s}^1\|_{F,2}^2 \leq J^2.$$

We have thus obtained that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\tilde{\Phi}_j^0(x)^2] \min_{i=1\dots q} |\Delta \mathfrak{s}_{ij}^1| = O_{\hat{\mathbb{P}}}(1/\sqrt{n})$, and so we can conclude that $\Theta = \Omega_{\hat{\mathbb{P}}}(1)$, uniformly in \mathcal{B}_J and in probability wrt the random initialisation $\hat{\mathbb{P}}$.

Now, we will show that $G = O_{\hat{\mathbb{P}}}(1)$. We have

$$G = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|X_j\|_2^3] \leq \frac{4}{n} \sum_{j=1}^n \mathbb{E}[\|\tilde{X}_j\|_2^3] + \frac{4}{n} \sum_{j=1}^n \mathbb{E}[\|\Delta X_j\|_2^3].$$

Let us write $X_j = U_j + V_j$, with $U_j = (\zeta_{ij}^1 \Phi_j^0(x))_{i=1\dots q}$ and $V_j = (\mathfrak{m}_{ij}^1(\Phi_j^0(x) - \mathbb{E}[\Phi_j^0(x)]))_{i=1\dots q}$. Then $\|\Delta X_j\|_2^3 \leq 4(\|\Delta U_j\|_2^3 + \|\Delta V_j\|_2^3)$.

First, denoting as ζ_j^1 and $\Delta \mathfrak{s}_j^1$ the vectors $(\zeta_{ij}^1)_{i=1\dots q}$ and $(\Delta \mathfrak{s}_{ij}^1)_{i=1\dots q}$, we can write

$$\Delta U_j = \Delta \Phi_j^0(x) \zeta_j^1 + \tilde{\Phi}_j^0(x) \Delta \mathfrak{s}_j^1 \odot \zeta_j^1 + \Delta \Phi_j^0(x) \Delta \mathfrak{s}_j^1 \odot \zeta_j^1,$$

where \odot represents the Hadamard product. Φ^0 and ζ^1 are independent and $\mathbb{E}[|\zeta|^3] = 2\sqrt{2/\pi} < 2$ for $\zeta \sim \mathcal{N}(0, 1)$, so we have

$$\mathbb{E}[\|\Delta U_j\|_2^3] \leq 54(q^{3/2} \mathbb{E}[|\Delta \Phi_j^0(x)|^3] + \mathbb{E}[|\tilde{\Phi}_j^0(x)|^3] \mathbb{E}[\|\Delta \mathfrak{s}_j^1\|_2^3] + \mathbb{E}[|\Delta \Phi_j^0(x)|^3] \mathbb{E}[\|\Delta \mathfrak{s}_j^1\|_2^3]).$$

Using that $\|\Delta \Phi^0(x)\|_3^3 \leq \|\Delta \Phi^0(x)\|_2^3 \leq C^3$, we have that

$$\frac{1}{n} \sum_{j=1}^n q^{3/2} \mathbb{E}[|\Delta \Phi_j^0(x)|^3] \leq \frac{q^{3/2} C^3}{n}$$

uniformly in \mathcal{B}_J . Then we can notice that $\|\Delta \mathfrak{s}_j^1\|_2 \leq \|\mathbf{1}_q\|_3 \|\Delta \mathfrak{s}_j^1\|_6 = q^{1/3} \|\Delta \mathfrak{s}_j^1\|_6$ by Hölder's inequality. Hence

$$\begin{aligned} \frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\tilde{\Phi}_j^0(x)|^3] \mathbb{E}[\|\Delta \mathfrak{s}_j^1\|_2^3] &\leq \frac{q \|\Delta \mathfrak{s}^1\|_{F,6}^3}{\sqrt{n}} \left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\tilde{\Phi}_j^0(x)|^6] \right)^{1/2} \\ &\leq \frac{qJ^3}{\sqrt{n}} \left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\tilde{\Phi}_j^0(x)|^6] \right)^{1/2} = O_{\hat{\mathbb{P}}}(1/\sqrt{n}) \end{aligned}$$

uniformly in \mathcal{B}_J , where the last equality comes from the usual argument that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\tilde{\Phi}_j^0(x)|^6]$ has a finite limit in probability (with respect to the random initialisation).

Finally, we can notice that $|\Phi_j^0(x)| \leq \|\Phi^0(x)\|_2 \leq C$ for all j , so that

$$\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\Delta \Phi_j^0(x)|^3] \mathbb{E}[\|\Delta \mathfrak{s}_j^1\|_2^3] \leq \frac{q^{1/2}}{n} C^3 \|\Delta \mathfrak{s}^1\|_{F,3}^3 \leq \frac{q^{1/2} C^3 J^3}{n}$$

uniformly in \mathcal{B}_J , where we used that $\|\Delta \mathfrak{s}_j^1\|_2 \leq \|\mathbf{1}_q\|_6 \|\Delta \mathfrak{s}_j^1\|_3 = q^{1/6} \|\Delta \mathfrak{s}_j^1\|_3$ by Hölder's inequality, and that $\|\Delta \mathfrak{s}^1\|_{F,3} \leq \|\Delta \mathfrak{s}^1\|_{F,2} \leq J$. We can hence conclude that

$$\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|\Delta U_j\|_2^3] = O_{\mathbb{P}}(1/\sqrt{n})$$

uniformly in \mathcal{B}_J .

Now we need to bound $\|\Delta V_j\|_2$. Letting $\mathbf{m}_j^1 = (\mathbf{m}_{ij}^1)_{i=1\dots q}$ and $\delta\Phi_j^0(x) = \Phi_j^0(x) - \mathbb{E}[\Phi_j^0(x)]$, it can be easily shown that

$$\|\Delta V_j\|_2 \leq |\delta\tilde{\Phi}_j^0(x)| \|\Delta \mathbf{m}_j^1\|_2 + |\Delta \delta\Phi_j^0(x)| \|\tilde{\mathbf{m}}_j^1\|_2 + \|\Delta \mathbf{m}_j^1\|_2 |\Delta \delta\Phi_j^0(x)|.$$

So we have

$$\begin{aligned} \frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|\Delta V_j\|_2^3] &\leq \frac{27}{n} \sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^3] \|\Delta \mathbf{m}_j^1\|_2^3 \\ &\quad + \frac{27}{n} \sum_{j=1}^n \mathbb{E}[|\Delta \delta\Phi_j^0(x)|^3] \|\tilde{\mathbf{m}}_j^1\|_2^3 + \frac{27}{n} \sum_{j=1}^n \mathbb{E}[|\Delta \delta\Phi_j^0(x)|^3] \|\Delta \mathbf{m}_j^1\|_2^3. \end{aligned}$$

Starting from the first term, we have that

$$\sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^3] \|\Delta \mathbf{m}_j^1\|_2^3 \leq \left(\sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^6] \right)^{1/2} \left(\sum_{j=1}^n \|\Delta \mathbf{m}_j^1\|_2^6 \right)^{1/2}.$$

From Hölder's inequality we have $\|\Delta \mathbf{m}_j^1\|_2 \leq \|\mathbf{1}_q\|_3 \|\Delta \mathbf{m}_j\|_6 = q^{1/3} \|\Delta \mathbf{m}_j\|_6$ and hence

$$\begin{aligned} \frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^3] \|\Delta \mathbf{m}_j^1\|_2^3 &\leq \frac{q \|\Delta \mathbf{m}^1\|_{F,6}^3}{\sqrt{n}} \left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^6] \right)^{1/2} \\ &\leq \frac{qJ^3}{\sqrt{n}} \left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^6] \right)^{1/2} = O_{\mathbb{P}}(1/\sqrt{n}) \end{aligned}$$

uniformly in \mathcal{B}_J , as $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\delta\tilde{\Phi}_j^0(x)|^6]$ tends in probability (wrt the random initialisation) to a finite limit.

Proceeding analogously, and noting that the L -Lipschitzianity of ϕ implies that $\mathbb{E}[|\Delta \delta\Phi_j^0(x)|^3] \leq 8L^3 \mathbb{E}[|\Delta Y_j^0(x)|^3]$, we get

$$\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\Delta \delta\Phi_j^0(x)|^3] \|\tilde{\mathbf{m}}_j^1\|_2^3 \leq \frac{8C^3}{\sqrt{n}} \left(\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|\tilde{\mathbf{m}}_j^1\|_2^6] \right)^{1/2} = O_{\mathbb{P}}(1/\sqrt{n})$$

uniformly in \mathcal{B}_J , and again we used the fact that $\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|\tilde{\mathbf{m}}_j^1\|_2^6]$ converges in probability (wrt the random initialisation) to a finite quantity to show that the above expression is of order $O_{\mathbb{P}}(1/\sqrt{n})$.

Finally, in a similar way we get

$$\frac{1}{n} \sum_{j=1}^n \mathbb{E}[|\Delta \delta \Phi_j^0(x)|^3] \|\Delta \mathbf{m}_j^1\|_2^3 \leq \frac{8qJ^3C^3}{n}$$

uniformly in \mathcal{B}_J . We can hence conclude that

$$\frac{1}{n} \sum_{j=1}^n \mathbb{E}[\|V_j\|_2^3] = O_{\mathbb{P}}(1/\sqrt{n})$$

and so that, as $n \rightarrow \infty$, $G \leq O_{\mathbb{P}}(1)$, uniformly in \mathcal{B}_J and in probability with respect to the random initialisation. This ends the proof. \blacksquare

Appendix B. Multiclass classification ($q > 2$)

In the framework of Section 4.1, things get more complicated when there are more than two classes. We can write

$$\mathbb{E}[\ell(\hat{f}(x), i^*)] = \mathbb{P}\left(F_{i^*}(x) \leq \max_{i \neq i^*} F_i(x)\right) = 1 - \mathbb{P}\left(F_{i^*}(x) > \max_{i \neq i^*} F_i(x)\right).$$

Hence, given a q -dimensional Gaussian vector $Y \sim \mathcal{N}(M, Q)$, we need to find an estimate of $\mathbb{P}(Y_{i^*} > \max_{i \neq i^*} Y_i)$.

The most trivial estimator would consist of sampling different realisations of Y and then give a MC estimate. However, as we are interested in the gradient of the expected loss, this method will not work. Indeed, the gradient of this estimate is the sum of the gradients of the 01-loss of each sample. As all these gradients are null, we do not obtain anything informative. We have thus to proceed in a less naive way.

Let us assume that $i^* = q$ (the largest label). Hence, we will focus on $\mathbb{P}(Y_q > \max_{i < q} Y_i)$. With a Cholesky-like algorithm, we can find a lower triangular matrix A such that $Y \sim AX + M$, where $X \sim \mathcal{N}(0, \text{Id})$. We have $Y_i = \sum_{i'=1}^q A_{ii'} X_{i'} + M_i$ and $A_{iq} = 0$ for $i = 1 \dots (q-1)$, while $A_{qq} > 0$. For $i < q$, we can write

$$\mathbb{P}(Y_q > Y_i) = \mathbb{P}\left(X_q > \sum_{i'=1}^{q-1} \frac{A_{ii'} - A_{qi'}}{A_{qq}} X_{i'} + \frac{M_i - M_q}{A_{qq}}\right).$$

Let us define the $(q-1)$ dimensional random vector \tilde{X} as $\tilde{X} = \tilde{A}X + \tilde{M}$, where \tilde{A} is a $(q-1) \times q$ matrix and \tilde{M} is a $(q-1)$ vector, whose elements are given by $\tilde{A}_{ii'} = \frac{A_{ii'} - A_{qi'}}{A_{qq}}$ and $\tilde{M}_i = \frac{M_i - M_q}{A_{qq}}$ respectively. With this notation, we have $\mathbb{P}(Y_q > Y_i) = \mathbb{P}(X_q > \tilde{X}_i)$. Now, we have gained that X_q is independent from all the other X_i 's, and so from all the \tilde{X}_i 's. In short, $(X_q | \tilde{X}) = X_q \sim \mathcal{N}(0, 1)$. So, we can write

$$\mathbb{P}\left(Y_q > \max_{i < q} Y_i\right) = \mathbb{P}\left(X_q > \max_{i < q} \tilde{X}_i\right) = \mathbb{E}\left[\mathbb{P}\left(X_q > \max_{i < q} \tilde{X}_i \mid \tilde{X}\right)\right].$$

Now, if we let $\psi(u) = \frac{1}{2}(1 - \operatorname{erf}(u/\sqrt{2}))$, we get

$$\mathbb{P}\left(Y_q > \max_{i < q} Y_i\right) = \mathbb{E}\left[\psi\left(\max_{i < q} \tilde{X}_i\right)\right].$$

We can estimate the above expression with MC sampling. Note that it is almost everywhere differentiable with respect to the components of M and Q (as the Cholesky transform is differentiable) and the gradient with respect to M and Q is not trivially null.

Finally, for a general $i^* \in \{1, \dots, q\}$, we can get $\mathbb{P}(Y_{i^*} > \max_{i \neq i^*} Y_i)$ by simply performing a swap of the two labels i^* and q , and then apply the method for $i^* = q$.

Appendix C. Expected values for ReLU and sin activations

Let $a > 0, b \in \mathbb{R}, \zeta \sim \mathcal{N}(0, 1)$. The following formulae are easily verified by direct calculations:

$$\begin{aligned} \mathbb{E}[\sin(a\zeta + b)] &= e^{-a^2/2} \sin b; \\ \mathbb{E}[\sin(a\zeta + b)^2] &= \frac{1}{2}(1 - e^{2a^2} \cos(2b)); \\ \mathbb{E}[\operatorname{ReLU}(a\zeta + b)] &= \frac{ae^{-b^2/(2a^2)}}{\sqrt{2\pi}} + \frac{b}{2} \left(1 + \operatorname{erf} \frac{b}{a\sqrt{2}}\right); \\ \mathbb{E}[\operatorname{ReLU}(a\zeta + b)^2] &= \frac{abe^{-b^2/(2a^2)}}{\sqrt{2\pi}} + \frac{1}{2}(a^2 + b^2) \left(1 + \operatorname{erf} \frac{b}{a\sqrt{2}}\right). \end{aligned}$$

Appendix D. Experimental details

In all the experiments, the training consisted of optimising some PAC-Bayesian bound via SGD with momentum parameter 0.9. The PAC parameter δ was always chosen equal to 0.025. We only performed the training of the means \mathbf{m} and all the networks considered had no bias. The priors corresponded to the initialisation of the network (6). Note that in our implementation, the scaling factors $1/\sqrt{p}$ and $1/\sqrt{n}$ were absorbed in the hyper-parameters, so that we performed the gradient descent on $\mu^0 = \mathbf{m}^0/\sqrt{p}$ and $\mu^1 = \mathbf{m}^1/\sqrt{n}$ (the standard deviations were kept fixed).

For the binary MNIST experiments, the digits from 0 to 4 were relabelled as 0 and those from 5 to 9 as 1. The training dataset used was the standard one for MNIST, consisting of $m = 60000$ datapoints. For the “standard” PAC-Bayesian methods, the objectives used are those reported in (11). For the objective 1bd we proceeded by alternating the optimisation of the network hyper-parameters with that of λ , as in Pérez-Ortiz et al. (2021), always enforcing $\lambda \in (0, 1)$. The “Gaussian” training was performed with the optimisation objective (12). All of these methods were used to train the same stochastic network, initialised as in (6). We tried two different learning rate (LR) schedules, the first consisting of 10000 epochs with LR $\eta = 10^{-5}$ and the second of 100 epochs with $\eta = 10^{-2}$, followed by 1000 epochs with $\eta = 10^{-3}$ and 5000 epochs with $\eta = 10^{-4}$. In Table 1 in the main text we report the results of the training schedule achieving the tightest bound, that is the multi-LR schedule for invk1 and quad, and the single-LR schedule for McAll and 1bd.

For the full MNIST experiments, again we used the standard training dataset with $m = 60000$ datapoints. For the “standard” methods, $L_S(\mathcal{Q})$ in (11) was a bounded version of the cross-entropy: we fixed $p_0 = 10^{-5}$ and constrained the probabilities in the definition of the cross-entropy to be greater or equal than p_0 , see (Pérez-Ortiz et al., 2021) for more details. In this way, the loss is

bounded by $\log(1/p_0)$, and by rescaling it of the same factor we can get a loss bounded in $[0, 1]$. $L_S(\mathcal{Q})$ is the empirical average of this quantity on the training dataset. As we previously did for the binary MNIST experiment, during the training we estimated $L_S(\mathcal{Q})$ by sampling once per iteration the hyper-parameters of the network. The “Gaussian” method used the objective (12), where $L_S(\mathcal{Q})$ is the expected empirical 01-loss. As we are dealing with multiclass classification we do not have a simple expression for the 01-loss, so we used the method described in Appendix B. Per each iteration, the loss was evaluated by an MC estimate averaging 10^4 independent realisations. For all the methods, the training consisted of 10000 epochs with learning rate $\eta = 10^{-5}$.

Conditionally Gaussian PAC-Bayes

Eugenio Clerico*

George Deligiannidis

Arnaud Doucet

Department of Statistics, University of Oxford

Abstract

Recent studies have empirically investigated different methods to train stochastic neural networks on a classification task by optimising a PAC-Bayesian bound via stochastic gradient descent. Most of these procedures need to replace the misclassification error with a surrogate loss, leading to a mismatch between the optimisation objective and the actual generalisation bound. The present paper proposes a novel training algorithm that optimises the PAC-Bayesian bound, without relying on any surrogate loss. Empirical results show that this approach outperforms currently available PAC-Bayesian training methods.

1 INTRODUCTION

Understanding generalisation for neural networks is among the most challenging tasks for learning theorists (Allen-Zhu et al., 2019; Kawaguchi et al., 2017; Neyshabur et al., 2017; Poggio et al., 2020; Zhang et al., 2021). Only a few of the theoretical tools developed in the literature can produce non-vacuous bounds on the error rates of over-parametrised architectures, and PAC-Bayesian bounds have proven to be among the tightest in the context of supervised classification (Ambroladze et al., 2007; Langford and Shawe-Taylor, 2003; McAllester, 2004). Several recent works have focused on algorithms aiming to minimise a generalisation bound for stochastic classifiers by optimising a PAC-Bayesian objective via stochastic gradient descent; see e.g. Alquier et al. (2016); Biggs and Guedj (2021); Clerico et al. (2021); Dziugaite and Roy (2017); Letarte et al. (2019); Pérez-Ortiz et al. (2021a,b); Zhou

et al. (2019). Most of these studies use a surrogate loss to avoid dealing with the zero-gradient of the misclassification loss. However, there are exceptions, such as Biggs and Guedj (2021) and Clerico et al. (2021), which rely on the fact that an analytically tractable output distribution allows for an estimate of the misclassification error with a non-zero gradient with respect to the trainable parameters of the classifier.

Clerico et al. (2021) treat the case of a stochastic network with a single hidden layer. They prove a Central Limit Theorem (CLT) ensuring the convergence of the output distribution to a multivariate Gaussian, whose mean and covariance can be evaluated explicitly in terms of the network deterministic hyper-parameters. However, this result cannot be straightforwardly extended to the multilayer case, as the nodes of the deeper layers are not independent and so the CLT might not apply. Moreover, even assuming that the output is Gaussian, the computational cost of this method is prohibitive for deep architectures.

In Biggs and Guedj (2021), the focus is on a stochastic binary classifier whose output is of the form $\text{sign}(w \cdot a)$, where w is a Gaussian vector and a is the output of the last hidden layer. The explicit form of the conditional expectation of the network’s output (conditioned with respect to a) allows for a PAC-Bayesian training method applicable to arbitrarily deep networks. Nevertheless, this approach is only suitable for binary classification and cannot be easily extended to the multiclass case.

In the present work, we conjugate the two above ideas: in order to train the network with a method inspired by the Gaussian PAC-Bayesian approach from Clerico et al. (2021), we exploit the output’s Gaussianity that can be obtained by conditioning on the previous layers, as in Biggs and Guedj (2021). This training procedure can be applied to a fairly general class of stochastic classifiers, overcoming some of the main limitations of the two aforementioned works, namely the single hidden layer and the binary classification setting. The

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151. Copyright 2022 by the author(s).

* Correspondence to: clerico@stats.ox.ac.uk

main requirement for our method to be valid is that the parameters of the last linear layer are independent Gaussian random variables. Additionally, as we are not relying on any CLT result to obtain the Gaussianity, we do not need the network to be very wide for the algorithm to work. Consequently, the approach we propose can be computationally much cheaper than the one from Clerico et al. (2021).

We empirically validate our training algorithm on MNIST and CIFAR10 for a range of architectures, testing both data-dependent and data-free PAC-Bayesian priors. We compare our results to those from Pérez-Ortiz et al. (2021a), as, to our knowledge, these are currently the tightest empirical PAC-Bayesian bounds available on these datasets. Our novel approach outperforms their standard PAC-Bayesian training methods in all our experiments.

2 BACKGROUND

2.1 PAC-Bayesian framework

In a standard classification problem, to each instance $x \in \mathcal{X} \subseteq \mathbb{R}^p$ corresponds a true label $y = f(x) \in \mathcal{Y} = \{1, \dots, q\}$. A training set $S = (X_k)_{k=1, \dots, m}$ is correctly labelled: for every $X_k \in S$ we have access to $Y_k = f(X_k)$. Each X_k is an independent draw from a fixed probability measure \mathbb{P}_X on \mathcal{X} , so that $S \sim \mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. We consider a neural network, namely a parameterised function $F^\theta : \mathbb{R}^p \rightarrow \mathbb{R}^q$. For each input x , the network returns a prediction \hat{y} , defined as the largest output’s node index:

$$\hat{y} = \hat{f}^\theta(x) = \operatorname{argmax}_{i \in \{1, \dots, q\}} F_i^\theta(x).$$

The goal is to train the net to make good predictions, exploiting the information in S to tune the parameters.

Define the misclassification loss as

$$\ell(\hat{y}, y) = \begin{cases} 0 & \text{if } y = \hat{y}; \\ 1 & \text{otherwise.} \end{cases} \quad (1)$$

For a given configuration θ of the network parameters, we call empirical error the empirical mean of the misclassification loss on the training sample: $\mathcal{E}_S(\theta) = \frac{1}{m} \sum_{x \in S} \ell(\hat{f}^\theta(x), f(x))$. This quantity can be explicitly evaluated, as we have access to the true labels on S . Therefore, it can be seen as an estimate for the true error $\mathcal{E}_\mathbb{P}(\theta) = \mathbb{E}_X[\ell(\hat{f}^\theta(X), f(X))] = \mathbb{P}_X(\hat{f}^\theta(X) \neq f(X))$, which in general cannot be computed exactly.

The PAC-Bayesian bounds are upper bounds on the true error, holding with high probability on the choice of the training sample S ; see e.g. Alquier (2021); Catoni (2007); Guedj (2019); McAllester (1998, 1999). A main feature of the PAC-Bayesian framework is that

it requires the network to be stochastic, that is we are dealing with architectures whose parameters θ are random variables.

Let us fix \mathcal{P} , a probability measure for the parameters θ . We assume that \mathcal{P} is data-independent, in the sense that it has to be selected without accessing the information in the training sample S . In line with most PAC-Bayesian literature, we will refer to \mathcal{P} as the prior distribution. For a stochastic network, the training consists in efficiently modifying the distribution of θ . This leads to a new distribution \mathcal{Q} on the parameters, usually referred to as the posterior distribution. The main idea behind the PAC-Bayesian theory is that if the posterior \mathcal{Q} is not “too far” from the prior \mathcal{P} , then the network should not be prone to overfitting. The essential tool to measure this “distance” between the prior and posterior distributions is the Kullback–Leibler divergence, defined as

$$\operatorname{KL}(\mathcal{Q} \parallel \mathcal{P}) = \begin{cases} \mathbb{E}_{\theta \sim \mathcal{Q}} \left[\log \frac{d\mathcal{Q}}{d\mathcal{P}}(\theta) \right] & \text{if } \mathcal{Q} \ll \mathcal{P}; \\ +\infty & \text{otherwise.} \end{cases}$$

The PAC-Bayesian bounds are upper bounds on the expected value of the true classification error $\mathcal{E}_\mathbb{P}$ with respect to the posterior \mathcal{Q} . Two main ingredients constitute these bounds: the expected empirical error under \mathcal{Q} and a complexity term, involving the divergence $\operatorname{KL}(\mathcal{Q} \parallel \mathcal{P})$. For simplicity, we will introduce the notations $\mathcal{E}_\mathbb{P}(\mathcal{Q}) = \mathbb{E}_{\theta \sim \mathcal{Q}}[\mathcal{E}_\mathbb{P}(\theta)]$ and $\mathcal{E}_S(\mathcal{Q}) = \mathbb{E}_{\theta \sim \mathcal{Q}}[\mathcal{E}_S(\theta)]$. The next proposition states some frequently used PAC-Bayes bounds (Langford and Seeger, 2001; Maurer, 2004; McAllester, 1999; Pérez-Ortiz et al., 2021a; Thiemann et al., 2017).

Proposition 1. Fix $\delta \in (0, 1)$, a data-independent prior \mathcal{P} , and a training set $S = (X_k)_{k=1, \dots, m}$ drawn according to \mathbb{P}_S . Define

$$\operatorname{Pen} = \frac{1}{m} (\operatorname{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}); \quad (2)$$

$$\operatorname{kl}^{-1}(u|c) = \sup\{v \in [0, 1] : \operatorname{kl}(u \parallel v) \leq c\}, \quad (3)$$

where $\operatorname{kl}(u \parallel v)$ denotes the KL divergence between two Bernoulli distributions, with means u and v respectively. Then, with probability at least $1 - \delta$ on the random draw of the training set, for any posterior \mathcal{Q} each of the following quantities upper bounds $\mathcal{E}_\mathbb{P}(\mathcal{Q})$:¹

$$\mathcal{B}_1 = \operatorname{kl}^{-1}(\mathcal{E}_S(\mathcal{Q}) \parallel \operatorname{Pen}); \quad (4a)$$

$$\mathcal{B}_2 = \mathcal{E}_S(\mathcal{Q}) + \sqrt{\operatorname{Pen}/2}; \quad (4b)$$

$$\mathcal{B}_3 = (\sqrt{\mathcal{E}_S(\mathcal{Q}) + \operatorname{Pen}/2} + \sqrt{\operatorname{Pen}/2})^2; \quad (4c)$$

$$\mathcal{B}_4 = \inf_{\lambda \in (0, 1)} \frac{1}{1-\lambda/2} (\mathcal{E}_S(\mathcal{Q}) + \operatorname{Pen}/\lambda). \quad (4d)$$

¹For (4a) we additionally assume that S has size $m \geq 8$.

In the above proposition, the bound \mathcal{B}_1 is always the tightest. Moreover, all the above bounds are still valid if the empirical classification error \mathcal{E}_S is replaced by the empirical average of any loss function $\tilde{\ell}$ in $[0, 1]$.

So far, we have assumed the prior \mathcal{P} to be data-independent. However, empirical evidence shows that using a data-dependent prior can lead to much tighter generalisation bounds, see e.g. Ambroladze et al. (2007); Dziugaite and Roy (2018); Dziugaite et al. (2021); Parrado-Hernández et al. (2012); Pérez-Ortiz et al. (2021b). Indeed, the actual requirement for the bounds (4) to hold is that \mathcal{P} is independent of the sample S used to evaluate $\mathcal{E}_S(\mathcal{Q})$. Hence, one can split the dataset S into two disjoint sets, $S^{(1)}$ and $S^{(2)}$, use $S^{(1)}$ to train the prior, and obtain the data-dependent versions of the PAC-Bayesian bounds from Proposition 1, by redefining $\text{Pen} = (\text{KL}(\mathcal{Q} \parallel \mathcal{P}_{S^{(1)}}) + \log \frac{2\sqrt{m_2}}{\delta})/m_2$ and replacing $\mathcal{E}_S(\mathcal{Q})$ with $\mathcal{E}_{S^{(2)}}(\mathcal{Q})$. For instance (4a) becomes

$$\mathcal{E}_{\mathbb{P}}(\mathcal{Q}) \leq \text{kl}^{-1} \left(\mathcal{E}_{S^{(2)}}(\mathcal{Q}) \left| \frac{\text{KL}(\mathcal{Q} \parallel \mathcal{P}_{S^{(1)}}) + \log \frac{2\sqrt{m_2}}{\delta}}{m_2} \right. \right), \quad (5)$$

where $m_2 \geq 8$ is the size of $S^{(2)}$.

2.2 PAC-Bayesian training

Ideally, one would like to implement the following procedure (McAllester, 1998):

- Fix the PAC parameter $\delta \in (0, 1)$ and a prior \mathcal{P} for the network stochastic parameters;
- Collect a sample S of m iid data points, according to $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$, and label it correctly;
- Compute an optimal posterior \mathcal{Q} minimising a generalisation bound, such as (4a);
- Implement a stochastic network whose random parameters have distribution \mathcal{Q} .

Unfortunately, in most realistic non-trivial scenarios, it can be extremely hard to compute and sample from an optimal posterior \mathcal{Q} (Guedj, 2019). A possible approach consists in using Markov chain Monte Carlo (Alquier and Biau, 2013; Dalalyan and Tsybakov, 2012; Guedj and Alquier, 2013), sequential Monte Carlo or variational methods (Alquier et al., 2016), in order to approximately sample from the Gibbs posterior, which can be shown to be the optimal \mathcal{Q} when the PAC-Bayesian bound is linear in the empirical loss (Catoni, 2007). However, these methods can often be inefficient, especially in the case of deep architectures and large datasets.

An alternative approach relies on simplifying the problem by constraining \mathcal{P} and \mathcal{Q} to belong to some simple distribution class. A common choice is to focus on the case of multivariate Gaussian distributions with

diagonal covariance (Dziugaite and Roy, 2017; Pérez-Ortiz et al., 2021a): all the parameters are independent normal random variables. Conveniently, in this case the law of the random parameters can be easily expressed in terms of their means and standard deviations. These are deterministic trainable quantities that we will call hyper-parameters and denote by \mathbf{p} . Furthermore, with this choice of \mathcal{P} and \mathcal{Q} , the KL divergence between prior and posterior takes a simple closed-form. Denoting as \mathbf{m} and \mathbf{s} (resp. $\tilde{\mathbf{m}}$ and $\tilde{\mathbf{s}}$) the means and standard deviations of the posterior (resp. prior), we have

$$\text{KL}(\mathcal{Q} \parallel \mathcal{P}) = \frac{1}{2} \sum_k \frac{\mathbf{s}_k^2 - \tilde{\mathbf{s}}_k^2}{\tilde{\mathbf{s}}_k^2} + \frac{1}{2} \sum_k \left(\frac{\mathbf{m}_k - \tilde{\mathbf{m}}_k}{\tilde{\mathbf{s}}_k} \right)^2 + \sum_k \log \frac{\tilde{\mathbf{s}}_k}{\mathbf{s}_k},$$

where the index k runs over all the stochastic parameters of the networks.

Now, the idea is to tune the hyper-parameters $\mathbf{p} = (\mathbf{m}, \mathbf{s})$ to minimise a PAC-Bayesian bound, such as (4a). A natural way to proceed is to perform a numerical optimisation via stochastic gradient descent, an approach originally proposed by Germain et al. (2009) and Dziugaite and Roy (2017), and referred to as PAC-Bayes with BackProp by Pérez-Ortiz et al. (2021a). First, we fix a PAC-Bayesian bound as our optimisation objective. As previously mentioned, this will be an expression involving a complexity term and the empirical error (Pen and $\mathcal{E}_S(\mathcal{Q})$ respectively). We will hence denote it as $\mathcal{B}(\mathcal{E}_S(\mathcal{Q}), \text{Pen})$. Generally, an explicit form for $\mathcal{E}_S(\mathcal{Q})$ is not available, but sampling from \mathcal{Q} easily provides an unbiased estimate $\hat{\mathcal{E}}_S(\mathcal{Q})$ of this quantity. However, we cannot perform a gradient descent step on $\mathcal{B}(\hat{\mathcal{E}}_S(\mathcal{Q}), \text{Pen})$. Indeed, $\hat{\mathcal{E}}_S(\mathcal{Q})$ has a null gradient almost everywhere, as it is the average over a finite set of realisations of the misclassification loss, which is constant almost everywhere (Pérez-Ortiz et al., 2021a). In order to overcome this problem, it is common to use a surrogate loss function (usually a bounded version of the cross-entropy) instead of the misclassification loss; see e.g. (Dziugaite and Roy, 2017) and (Pérez-Ortiz et al., 2021a,b). However, this creates a mismatch between the optimisation objective and the actual target bound.

It is worth noting that the zero-gradient problem is due to the particular form of the estimate $\hat{\mathcal{E}}_S(\mathcal{Q})$ and in general $\mathcal{E}_S(\mathcal{Q})$ has a non-zero gradient (Clerico et al., 2021). Indeed, as it will be shown in Section 3, a different choice of estimator for $\mathcal{E}_S(\mathcal{Q})$ can allow training the network without the use of any surrogate loss.

2.3 Stochastic network and notations

Consider a stochastic classifier featuring several hidden layers and a final linear layer. We denote $H(x)$ the output of the last hidden layer when the input is x , ϕ the activation function (here applied component-wise), and W and B the weight and bias of the linear output layer. The output of the network will be

$$F(x) = W\phi(H(x)) + B, \quad (6)$$

where we wrote F instead of F^θ to simplify the notation. Since the network is stochastic, W , B , and $H(x)$ are random quantities. We denote $\mathcal{F}^\mathcal{L}$ the σ -algebra generated by the last layer's stochasticity, and $\mathcal{F}^\mathcal{H}$ the one due to the hidden layers.

We will henceforth assume the following:

- $\mathcal{F}^\mathcal{L} \perp\!\!\!\perp \mathcal{F}^\mathcal{H}$, that is the two σ -algebras are independent;
- W and B have independent normal components.

We can thus express the stochastic parameters of the last layer in terms of a set of deterministic trainable hyper-parameters \mathbf{m} and \mathbf{s} :

$$W_{ij} = \zeta_{ij}^W \mathbf{s}_{ij}^W + \mathbf{m}_{ij}^W; \quad B_i = \zeta_i^B \mathbf{s}_i^B + \mathbf{m}_i^B,$$

where the ζ are all independent standard normal random variables $\mathcal{N}(0, 1)$.

For the hidden layers, we do not require any strong assumption: essentially, we need to be able to sample a realisation $h(x)$ of $H(x)$, to evaluate the KL divergence between prior and posterior, and to differentiate both KL and $h(x)$ with respect to the trainable deterministic hyper-parameters. However, for the sake of simplicity, in the rest of this paper we will assume that all the parameters of the hidden layers have independent normal laws, as in Clerico et al. (2021); Dziugaite and Roy (2017); Pérez-Ortiz et al. (2021a). All the architectures used for our experiments are indeed in this form. We refer to the supplementary material (Section SM3) for the extension of our results on more general architectures.

3 COND-GAUSS ALGORITHM

We present here a training procedure to optimise a PAC-Bayesian generalisation bound without the need for a surrogate loss. The two main ideas are the following:

- An unbiased estimate of $\mathcal{E}_S(\mathcal{Q})$ and its gradient can be evaluated if the output of the network is Gaussian, as in Clerico et al. (2021);
- If the parameters of the last layer are Gaussian, the output of the network is Gaussian as well

when conditioned on the nodes of the last hidden layer, as pointed out by Biggs and Guedj (2021).

3.1 Gaussian output

Fix an input x and assume that the network's output $F(x)$ follows a multivariate normal distribution, with mean vector $M(x)$ and covariance matrix $Q(x)$. For our purposes, we can suppose that $Q(x)$ is diagonal, meaning that the components of the output are mutually independent (we refer to Section 4.1 in Clerico et al. (2021) for the discussion of the general case). Let us denote $V(x)$ the diagonal of $Q(x)$, consisting of the output's variances, so that

$$\mathbb{E}_{\mathcal{Q}}[F_i(x)] = M_i(x); \quad \mathbb{V}_{\mathcal{Q}}[F_i(x)] = V_i(x).$$

The stochastic prediction of our classifier is $\hat{y} = \hat{f}(x) = \operatorname{argmax}_{i \in \{1, \dots, q\}} F_i(x)$. In order to compute $\mathcal{E}_S(\mathcal{Q})$, for each input $x \in S$ we shall evaluate $\mathbb{E}_{\mathcal{Q}}[\ell(\hat{f}(x), f(x))]$. As ℓ is the misclassification loss (1), this is simply the probability of making a mistake for the input x . Letting $y = f(x)$ and $\hat{y} = \hat{f}(x)$, we have

$$\mathbb{E}_{\mathcal{Q}}[\ell(\hat{y}, y)] = \mathbb{P}_{\mathcal{Q}}(\hat{y} \neq y) = \mathbb{P}_{\mathcal{Q}}\left(F_y(x) \leq \max_{i \neq y} F_i(x)\right). \quad (7)$$

In the case of binary classification, the above expression has a simple closed-form. Indeed, if we consider for instance the case $y = 1$, we have

$$\begin{aligned} \mathbb{P}_{\mathcal{Q}}(\hat{y} \neq 1) &= \mathbb{P}_{\mathcal{Q}}(F_2(x) - F_1(x) \geq 0) \\ &= \mathbb{P}\left(\zeta \leq \frac{M_2(x) - M_1(x)}{\sqrt{V_1(x) + V_2(x)}}\right), \end{aligned}$$

where $\zeta \sim \mathcal{N}(0, 1)$. This can be expressed in terms of the error function erf , as the cumulative distribution function of a standard normal is given by $\psi(u) = \mathbb{P}(\zeta \leq u) = \frac{1}{2}(1 + \operatorname{erf}(u/\sqrt{2}))$. Notice that the above expression no longer suffers from vanishing gradients, as $\psi' \neq 0$.

For multiple classes ($q > 2$), (7) does not have a simple closed-form. However, we can easily find Monte Carlo estimators that also bring unbiased estimates for the gradient with respect to M and Q .

Proposition 2. *Denote the cumulative distribution function of a standard normal random variable as $\psi : u \mapsto \frac{1}{2}(1 + \operatorname{erf}(u/\sqrt{2}))$. Fix x , let y be its true label, and \hat{y} the network's stochastic prediction. Define*

$$\begin{aligned} L_1 &= \psi\left(\max_{i \neq y} \frac{F_i(x) - M_y(x)}{\sqrt{V_y(x)}}\right); \\ L_2 &= 1 - \prod_{i \neq y} \psi\left(\frac{F_y(x) - M_i(x)}{\sqrt{V_i(x)}}\right), \end{aligned}$$

where $F(x) \sim \mathcal{N}(M(x), \text{diag}(V(x)))$. Then

$$\begin{aligned}\mathbb{E}_{\mathcal{Q}}[L_1] &= \mathbb{E}_{\mathcal{Q}}[L_2] = \mathbb{P}_{\mathcal{Q}}(\hat{y} \neq y), \\ \mathbb{E}_{\mathcal{Q}}[\nabla L_1] &= \mathbb{E}_{\mathcal{Q}}[\nabla L_2] = \nabla \mathbb{P}_{\mathcal{Q}}(\hat{y} \neq y),\end{aligned}$$

where the gradient is with respect to all the components of $M(x)$ and $V(x)$.

In particular, by sampling realisations of L_1 or L_2 , we can get unbiased Monte Carlo estimators of the misclassification loss and its gradient.

3.2 Conditional Gaussianity

In practice, the output of a stochastic network is generally not Gaussian. However, we can overcome this issue by conditioning on the hidden layers, similarly to what was done by Biggs and Guedj (2021).

Recall that the network's output is given by (6):

$$F = W\phi(H) + B,$$

where the explicit dependence of H on x is omitted to make the notations lighter. Conditioned on the stochasticity of the hidden layers \mathcal{F}^H , F follows a normal multivariate distribution, as

$$F = W\phi(H) + B \sim \mathcal{N}(M(H), Q(H)).$$

We can easily evaluate $M(H)$ and $Q(H)$ in terms of \mathbf{m} and \mathbf{s} . We have

$$\begin{aligned}M_i(H) &= \mathbb{E}_{\mathcal{Q}}[F_i | \mathcal{F}^H] = \sum_j \mathbb{E}_{\mathcal{Q}}[W_{ij}] \phi(H_j) + \mathbb{E}_{\mathcal{Q}}[B_i] \\ &= \sum_j \mathbf{m}_{ij}^W \phi(H_j) + \mathbf{m}_i^B\end{aligned}$$

and $Q_{ij}(H) = \delta_{ij}V_i(H)$, with

$$\begin{aligned}V_i(H) &= \mathbb{V}_{\mathcal{Q}}[F_i | \mathcal{F}^H] = \sum_j \mathbb{V}_{\mathcal{Q}}[W_{ij}] \phi(H_j)^2 + \mathbb{V}_{\mathcal{Q}}[B_i] \\ &= \sum_j (\mathbf{s}_{ij}^W \phi(H_j))^2 + (\mathbf{s}_i^B)^2.\end{aligned}$$

Finally, we note that by iterated expectations

$$\mathbb{E}_{\mathcal{Q}}[\ell(\hat{f}(x), f(x))] = \mathbb{E}_{\mathcal{Q}}[\mathbb{E}_{\mathcal{Q}}[\ell(\hat{f}(x), f(x)) | \mathcal{F}^H]].$$

In particular, if we draw the hidden parameters and get a realisation h of H , we obtain an unbiased estimate $\frac{1}{m} \sum_{x \in S} \mathbb{E}[\ell(\hat{f}(x), f(x)) | H(x) = h(x)]$ of $\mathcal{E}_S(\mathcal{Q})$, where each term $\mathbb{E}[\ell(\hat{f}(x), f(x)) | H(x) = h(x)]$ can be estimated with the methods from Section 3.1, since $F(x)$ is a multivariate Gaussian when conditioned on $H(x) = h(x)$.

3.3 Training algorithm

We sketch here the Cond-Gauss training algorithm. First, we fix a PAC-Bayesian bound \mathcal{B} as the optimisation objective. Then, we initialise the deterministic hyper-parameters of our network, and we select this configuration as the prior. Finally, we split our dataset into batches S_1, \dots, S_K . To train the network, we iterate over the batches and, similarly to what is done in most PAC-Bayesian training methods based on stochastic gradient descent, we sample the network's parameters at each batch iteration. However, we only perform this sampling for the hidden layers and not for the final linear layer. In this way, for each x in the batch, we have a realisation $h(x)$ of the last hidden layer's output. Conditioned on $H = h$, the output is

Algorithm 1 Cond-Gauss PAC-Bayesian training

Require:

$$\begin{aligned}\tilde{\mathbf{p}} &= (\tilde{\mathbf{p}}^H, \tilde{\mathbf{p}}^L) \\ S & \\ \delta &\in (0, 1) \\ \eta, T &\end{aligned}$$

- ▷ Initial hyper-parameters (defining the prior)
- ▷ Training set of size $\#S$
- ▷ PAC parameter
- ▷ Learning rate and number of epochs

Ensure:

Optimal \mathbf{p} parameterizing the posterior

1: **procedure** COND-GAUSS

2: $\mathbf{p}^H \leftarrow \tilde{\mathbf{p}}^H$

3: $\mathbf{p}^L = (\mathbf{m}, \mathbf{s}) \leftarrow \tilde{\mathbf{p}}^L$

4: **for** $t \leftarrow 1 : T$ **do**

5: Sample $\theta^H \sim \mathcal{Q}_{\mathbf{p}^H}^H$

▷ Sample the parameters of the hidden layers

6: $h = h(S, \theta^H)$

▷ Evaluate the last hidden layer's output for all $x \in S$

7: $M = M(h, \mathbf{m}) = \mathbf{m}^W \phi(h) + \mathbf{m}^B$

▷ Evaluate the conditional mean of the output

8: $V = V(h, \mathbf{s}) = (\mathbf{s}^W \phi(h))^2 + (\mathbf{s}^B)^2$

▷ Evaluate the conditional variance of the output

9: $\hat{\mathcal{E}}_S(\mathcal{Q}_{\mathbf{p}}) = \mathcal{E}(M, V)$

▷ Evaluate $\hat{\mathcal{E}}_S(\mathcal{Q}_{\mathbf{p}})$ from M and V as in Section 3.1

10: $\hat{\mathcal{B}} = \mathcal{B}(\hat{\mathcal{E}}_S(\mathcal{Q}_{\mathbf{p}}), \text{Pen})$

▷ Evaluate the estimate $\hat{\mathcal{B}}$ of the PAC-Bayesian objective \mathcal{B}

11: $\mathbf{p} \leftarrow \mathbf{p} - \eta \nabla_{\mathbf{p}} \hat{\mathcal{B}}$

▷ Perform the gradient step

12: **return** \mathbf{p}

Gaussian and we can proceed as discussed earlier to get an estimate $\hat{\mathcal{E}}_{S_k}(\mathcal{Q}, h)$ of $\mathcal{E}_S(\mathcal{Q})$. After that, we can obtain an estimate $\hat{\mathcal{B}}$ of the target bound \mathcal{B} , by replacing $\mathcal{E}_S(\mathcal{Q})$ with $\hat{\mathcal{E}}_{S_k}(\mathcal{Q}, h)$. Finally, we compute the gradient of $\hat{\mathcal{B}}$ with respect to the trainable hyper-parameters, and we perform the gradient step.

If we want to use a data-dependent prior, we simply split the dataset into two subsets $S^{(1)}$ and $S^{(2)}$, and then use $S^{(1)}$ to learn \mathcal{P} . For instance, we might train the prior using $\hat{\mathcal{E}}_{S^{(1)}}(\mathcal{Q})$ as optimisation objective or tuning only the prior’s means by treating the network as if it was deterministic, similarly to what was done in Pérez-Ortiz et al. (2021a). Once the prior’s training is complete, we perform the Cond-Gauss algorithm, replacing S with $S^{(2)}$.

The training procedure is summarised in Algorithm 1, where, for the sake of simplifying the notation, it is assumed that the whole training set forms a single batch. For convenience, we introduce the superscripts \mathcal{H} and \mathcal{L} to refer to the hidden layers and the last layer, respectively. Thus, we denote as $\theta = (\theta^{\mathcal{H}}, \theta^{\mathcal{L}})$ the random parameters of the network, where $\theta^{\mathcal{H}}$ are the parameters in the hidden layers, while $\theta^{\mathcal{L}} = (W, B)$ are those of the last layer. Similarly, $\mathbf{p}^{\mathcal{H}}$ are the deterministic hyper-parameters relative to the hidden layers, whilst $\mathbf{p}^{\mathcal{L}} = (\mathbf{m}, \mathbf{s})$ are those of the last layer. We introduced the subscript \mathbf{p} for the posterior \mathcal{Q} , to stress the fact that it is determined by the hyper-parameters, and we denoted by $\mathcal{Q}^{\mathcal{H}}$ the marginal posterior distribution for the hidden layers. Finally, the tilde notation represents the values at initialisation.

As a final remark, kl^{-1} is currently not implemented in most of the standard deep learning libraries. Yet, it can be easily computed numerically with few iterations of Newton’s method, as in Dziugaite and Roy (2017). Nevertheless, most of the empirical studies on PAC-Bayesian gradient descent optimisation (see e.g. Dziugaite and Roy (2017) and Pérez-Ortiz et al. (2021a)), do not use as objective (4a), in order to avoid computing ∇kl^{-1} . However, since this gradient can be expressed as a function of kl^{-1} itself, we were able to optimise (4a) in our experiments (see Section SM4 in the supplementary material for further details).

3.4 Unbiasedness of the estimates

One might wonder whether the estimates of \mathcal{B} and its gradient are actually unbiased. Notably, this is indeed the case if the chosen PAC-Bayesian objective \mathcal{B} is an affine function of the empirical error, as (4b) and (4d).

Proposition 3. *Assume that \mathcal{B} is locally Lipschitz in the hidden stochastic parameters $\theta^{\mathcal{H}}$, and that $\nabla_{\theta^{\mathcal{H}}}\mathcal{B}$*

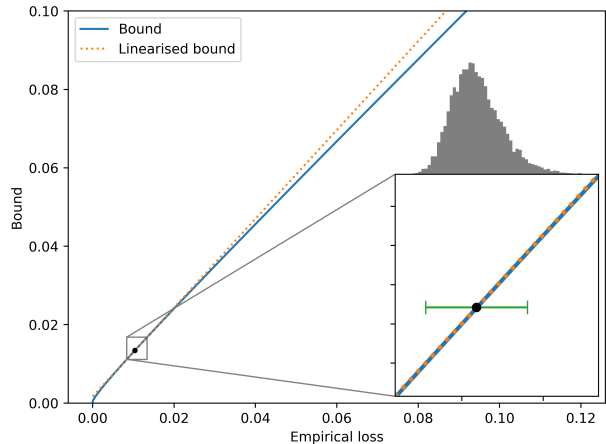


Figure 1: Experimental evidence that the bound (4a) is almost affine in the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates. The network used was the one achieving the best generalisation bound in our experiment on MNIST with *data-dependent* priors. 10000 realisations of $\hat{\mathcal{E}}_S(\mathcal{Q})$ were sampled. Their distribution is summarized by the histogram above the zoomed portion of the plot. The black dot is the bound for the average value found for $\hat{\mathcal{E}}_S(\mathcal{Q})$, while the green error bar has a total width of 4 empirical standard deviations. In the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates, the bound and its linearised version almost coincide. Along the green error bar, the bound’s slope has a relative variation of $\pm 0.8\%$.

is polynomially bounded.² If $\mathcal{B}(\mathcal{E}_S(\mathcal{Q}), \text{Pen})$ is affine in $\mathcal{E}_S(\mathcal{Q})$, then we have $\mathbb{E}[\hat{\mathcal{B}}] = \mathcal{B}$ and $\mathbb{E}[\nabla \hat{\mathcal{B}}] = \nabla \mathcal{B}$, the gradient being with respect to the trainable hyper-parameters \mathbf{p} .

Although this unbiasedness property does not hold for objectives not affine in $\mathcal{E}_S(\mathcal{Q})$, if $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates enough around $\mathcal{E}_S(\mathcal{Q})$ we can linearise $\hat{\mathcal{B}}$ as

$$\hat{\mathcal{B}} \simeq \mathcal{B} + (\hat{\mathcal{E}}_S(\mathcal{Q}) - \mathcal{E}_S(\mathcal{Q})) \partial_{\mathcal{E}} \mathcal{B}.$$

Then, both $\hat{\mathcal{B}}$ and $\nabla \hat{\mathcal{B}}$ are essentially almost unbiased estimates. Considering the good performance of our method in the experiments we ran, we conjecture that this is indeed what happens in practice with (4a) and (4c). Figure 1 gives some empirical support to this hypothesis. We refer the reader to the supplementary material (Section SM2) for additional discussion and empirical evidence on this subject.

3.5 Final evaluation of the bound

In order to evaluate the final generalisation bound, we need the exact value of $\mathcal{E}_S(\mathcal{Q})$ once the training is complete. As this cannot be computed, we use an empirical upper bound, as done for instance in (Dziugaite and Roy, 2017).

²These are mild technical assumptions, verified in all the experimental settings considered in this paper.

Let $\theta_1, \dots, \theta_N$ be N independent realisations of the whole set of the network stochastic parameters, drawn according to \mathcal{Q} . An unbiased Monte Carlo estimator of $\mathcal{E}_S(\mathcal{Q})$ is simply given by

$$\tilde{\mathcal{E}}_S(\mathcal{Q}) = \frac{1}{N} \sum_{n=1}^N \mathcal{E}_S(\theta_n).$$

As shown by Langford and Caruana (2002), for fixed $\delta' \in (0, 1)$, with probability at least $1 - \delta'$ we have,

$$\mathcal{E}_S(\mathcal{Q}) \leq \text{kl}^{-1} \left(\tilde{\mathcal{E}}_S(\mathcal{Q}) \middle| \frac{1}{N} \log \frac{2}{\delta'} \right),$$

where kl^{-1} is defined in (3). We conclude from Proposition 1 that, with probability higher than $1 - (\delta + \delta')$, we have

$$\begin{aligned} \mathcal{E}_{\mathbb{P}}(\mathcal{Q}) &\leq \text{kl}^{-1} \left(\text{kl}^{-1} \left(\tilde{\mathcal{E}}_S(\mathcal{Q}) \middle| \frac{1}{N} \log \frac{2}{\delta'} \right) \middle| \frac{\text{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta}}{m} \right), \end{aligned} \quad (8)$$

as kl^{-1} is an increasing function of its first argument.

4 NUMERICAL RESULTS

We tested the Cond-Gauss algorithm empirically on the MNIST and the CIFAR10 datasets (Deng, 2012; Krizhevsky, 2009). In the literature, several works benchmark various PAC-Bayesian algorithms on these and other datasets (Biggs and Guedj, 2021; Clerico et al., 2021; Dziugaite and Roy, 2017, 2018; Letarte et al., 2019; Pérez-Ortiz et al., 2021a,b). To our knowledge, in the case of over-parameterised deep neural networks, the bounds from Pérez-Ortiz et al. (2021a) are currently the tightest on both MNIST and CIFAR10. Thus, in order to assess our Cond-Gauss method by comparing their results with ours, we decided to mimic some of their multilayer convolutional architectures³, although our training schedules, as well as the prior’s training procedures and the choice of initial variances, differed from theirs. All the generalisation bounds obtained with our training algorithm were tighter than those reported by Pérez-Ortiz et al. (2021a).

We illustrate below some of our main empirical results. All the final generalisation bounds are obtained from (8), or its natural variant based on (5) for data-dependent priors. We always use $\delta = 0.025$ and $\delta' = 0.01$ as in Pérez-Ortiz et al. (2021a), so that

³The only difference between their architectures and ours is that we sometimes swapped the order between the application of the activation function and the max pooling. This fact was merely accidental, but we believe that it did not significantly affect our results.

the final generalisation bounds hold with probability greater or equal to 0.965. For all the bounds but those in Figure 2, we fixed $N = 150000$ as in Pérez-Ortiz et al. (2021a).

We refer to Section SM5 in the supplementary material for the full results and the missing experimental details. The PyTorch code developed for this paper is available at <https://github.com/eclerico/CondGauss>.

4.1 MNIST

For our experiments on MNIST, we only used the standard training dataset (60000 labelled examples) for the training procedure. We tested a 4-layer ReLU stochastic network, whose parameters were independent Gaussians. The architecture was composed of two convolutional layers followed by two linear layers.

We first experimented on data-free priors. We compared the performance of the standard PAC-Bayes with BackProp training algorithm (S), where the misclassification loss is replaced by a bounded version of the cross-entropy loss as in Pérez-Ortiz et al. (2021a), and the Cond-Gauss algorithm (G). We used the four training objectives from (4):

$$\begin{aligned} \text{invKL} : & \quad \text{kl}^{-1}(\mathcal{E}_S(\mathcal{Q}) \mid \text{Pen}_{\kappa}); \\ \text{McAll} : & \quad \mathcal{E}_S(\mathcal{Q}) + \sqrt{\text{Pen}_{\kappa}/2}; \\ \text{quad} : & \quad (\sqrt{\mathcal{E}_S(\mathcal{Q}) + \text{Pen}_{\kappa}/2} + \sqrt{\text{Pen}_{\kappa}/2})^2; \\ \text{1bd} : & \quad \frac{1}{1-\lambda/2} (\mathcal{E}_S(\mathcal{Q}) + \text{Pen}_{\kappa}/\lambda), \end{aligned}$$

where the KL penalty is defined as

$$\text{Pen}_{\kappa} = \frac{\kappa}{m} \left(\text{KL}(\mathcal{Q} \parallel \mathcal{P}) + \log \frac{2\sqrt{m}}{\delta} \right). \quad (9)$$

The factor κ in (9) can increase or reduce the weight of the KL term during the training. For the last objective, 1bd, the parameter λ takes values in $(0, 1)$ and is optimised during training, similarly to what was done in Pérez-Ortiz et al. (2021a).⁴

The network was trained via SGD with momentum. During training, at the end of each epoch, we kept track of the bound (4a)’s empirical value to pick the best epoch at the end of the training.

In Figure 2 we report the values of the bounds for different training settings with data-free priors on MNIST. As evaluating the true bound via (8) can be extremely time-consuming when $N = 150000$, the values reported in the figure are obtained for $N = 10000$.

⁴In our experiments, we initialised λ at 0.5 and then doubled the number of epochs, alternating one epoch of λ ’s optimisation with one of optimisation for m and s .

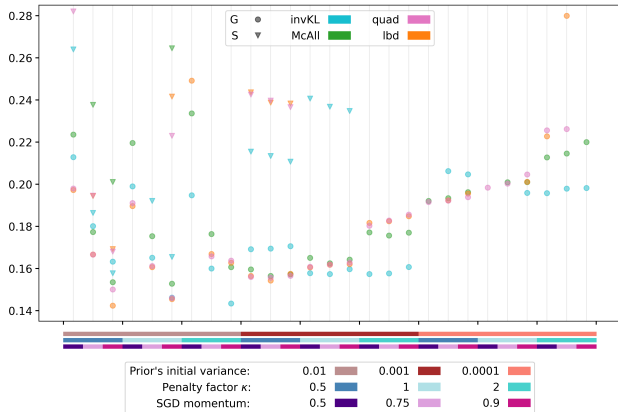


Figure 2: Results for MNIST with random prior. Each dot is the PAC-Bayesian bound obtained via (8) with $N = 10000$. The marker shape represents the training method (in the legend, ‘G’ stands for our method, ‘S’ for the standard one), and the colour represents the training objective. Different columns indicate different momentum values, penalty factor κ , and initial variance for the prior. The initial prior’s means were the same for all the different training possibilities. The values higher than 0.285 are not reported.

The Cond-Gauss algorithm always achieved the best performance. Note that some of the bounds in the figure are substantially tighter than the best value reported in Pérez-Ortiz et al. (2021a), namely .2165. Perhaps unexpectedly, this is sometimes the case even for the standard PAC-Bayes with BackProp algorithm, although it happened for training settings that were not tried therein, namely with the invKL objective or $\kappa = 0.5$.

In Table 1 we report the final generalisation bounds with $N = 150000$, evaluated via (8). For each method, we selected the training achieving the best bound in Figure 2. The Cond-Gauss procedure achieved better results than the standard algorithm with all the objectives. Quite surprisingly, the tightest bound was achieved by the lbd objective. The column ‘emp err’ reports the empirical error on the training dataset, ob-

Table 1: PAC-Bayesian bounds for MNIST - data-free prior

method	emp err	test err	Pen (2)	bound
S McAll	.0670	.0900 \pm .0047	.0320	.1916
S lbd	.0636	.0623 \pm .0013	.0413	.1606
S quad	.0622	.0577 \pm .0031	.0420	.1594
S invKL	.0438	.0407 \pm .0022	.0560	.1495
G McAll	.0472	.0435 \pm .0024	.0477	.1446
G lbd	.0279	.0272 \pm .0016	.0669	.1348
G quad	.0399	.0374 \pm .0021	.0518	.1380
G invKL	.0356	.0340 \pm .0019	.0556	.1355

tained when computing the final bounds. The test errors provided in the column ‘test err’ are evaluated on the standard held-out test dataset of MNIST, by averaging over 1000 realisations of the random network’s parameter. We also report the empirical standard deviation of this estimate. Interestingly, the test error on the held-out dataset often resulted smaller than the empirical error on the training dataset. We do not have an explanation for this fact, which might be a mere coincidence and did not occur in most of the experiments with data-dependent priors.

For the data-dependent priors, we used 50% of the dataset to train \mathcal{P} and the remaining 50% to train \mathcal{Q} . We always used the Cond-Gauss algorithm for both prior and posterior. All the posteriors were trained with the invKL objective and $\kappa = 1$, whilst for the prior, we experimented with different objectives, penalty factors κ , and dropout values. The final best generalisation bound was .0144, about 7% better than the tightest one from Pérez-Ortiz et al. (2021a) for the same architecture, .0155. However, it is interesting to note that the role of the posterior’s training seems to be quite marginal, as, in our experiments, the prior already achieved a quite low empirical error on the posterior’s dataset, .0108, which could be improved only to .0104 by tuning the posterior. The results of the whole experiment can be found in Table SM1 in the supplementary material.

4.2 CIFAR10

As we had done for the MNIST dataset, for CIFAR10 we used only the standard training dataset (50000 labelled images) for the training procedure. We trained a 9-layer architecture (6 convolutional + 3 linear layers) and a 15-layer architecture (12 convolutional + 3 linear layers). We experimented with data-dependent priors only, training \mathcal{P} with 50% of the data for the 9-layer classifier, and with both 50% and 70% of the data in the case of the 15-layer one.

The results for the 9-layer architecture are reported in Table 2. Note that the best bound that we obtained in this setting was .2066, a result much tighter than the one reported by Pérez-Ortiz et al. (2021a), .2901. After some preliminary experiments, we chose to train both priors and posteriors via the Cond-Gauss algorithm with the invKL objective. We used a small factor κ for the prior to avoid regularising too much, whilst κ was 1 for the posterior. We tried different values for the dropout and the factor κ in the training of the prior, as reported in Table 2. We trained via SGD with momentum for both prior and posterior. For \mathcal{P} , we used a schedule much longer than the one usually chosen for the prior in the literature. Essentially, this is because we were not just training the means of the

Table 2: CIFAR10 - 9 layers - Prior learnt on 50% of the dataset

Prior							Posterior					
tm ^a	do ^b	pf ^c	iv ^d	l1 ^e	l2 ^f	p ^g	tm ^a	l1 ^e	l2 ^f	t ^h	p ^g	b ⁱ
G invKL	0	.01	.001	.0196	.2233	3.778	G invKL	.0196	.2211	.2251 \pm .0021	4.696	.2376
G invKL	0	.005	.001	.1127	.2797	3.778	G invKL	.1126	.2782	.2814 \pm .0019	4.319	.2953
G invKL	.1	.01	.001	.0536	.1930	3.778	G invKL	.0536	.1912	.1952 \pm .0020	4.484	.2066
G invKL	.1	.005	.001	.0266	.1930	3.778	G invKL	.0266	.1913	.1933 \pm .0019	4.520	.2067

^a tm: Training method.^b do: Dropout probability for the prior’s training.^c pf: Penalty factor κ for the prior’s training objective.^d iv: Initial value of the prior’s variances.^e l1: Empirical error estimate on the prior dataset.^f l2: Empirical error estimate on the posterior dataset.^g t: Test error \pm standard deviation (from 1000 realisations).^h p: KL penalty **Pen (2)** in 10^{-4} units.ⁱ b: Final PAC-Bayesian bound.

random parameters, but the variances as well. However, in this way we could already obtain for the priors competitive empirical errors on the posterior’s dataset. Like with the MNIST dataset, the improvement due to the posterior’s training was minimal.

For the 15-layer architecture, the full results and details are reported in Table SM2 in the supplementary material. Quite interestingly, to train \mathcal{P} , it was necessary to introduce an initial pre-training for the prior’s means, as the Cond-Gauss algorithm alone could not significantly decrease the training objective. First, we initialised the means with an orthogonal initialisation, as suggested in Hu et al. (2020). Then we optimised them by training a deterministic network (with the same architecture) using the cross-entropy loss on the prior’s dataset. Finally, via the Cond-Gauss algorithm, we completed the prior’s training and proceeded with the posterior’s tuning. The best final bounds obtained were .1855, with the prior learnt on 50% of the dataset, and .1595, when 70% of the dataset was used to train \mathcal{P} . Again, these values are tighter than those from Pérez-Ortiz et al. (2021a).

4.3 Summary

To summarise our results, Table 3 compares our best PAC-Bayesian generalisation bounds with those from Pérez-Ortiz et al. (2021a). The column ‘C-G’ features the best bounds we could obtain with the Cond-Gauss

Table 3: Comparison of our PAC-Bayesian bounds with those from Pérez-Ortiz et al. (2021a)

dataset	architecture	prior	C-G	P-O
MNIST	4 layers	data-free	.1348	.2165
MNIST	4 layers	50%	.0144	.0155
CIFAR10	9 layers	50%	.2066	.2901
CIFAR10	15 layers	50%	.1855	.1954
CIFAR10	15 layers	70%	.1595	.1667

algorithm in our experiments. The figures in the column ‘P-O’ are the tightest bounds reported in Pérez-Ortiz et al. (2021a) for the same architectures and datasets. All the PAC-Bayesian generalisation bounds in the table hold with probability at least 0.965 on the choice of the training dataset.

5 CONCLUSION

We have introduced the Cond-Gauss training algorithm, which allows the optimisation of PAC-Bayesian bounds without relying on the use of a surrogate loss. Taking an estimate of the actual target bound as the optimisation objective is a natural choice. As confirmed by our experiments on the MNIST and the CIFAR10 classification tasks, it also leads to tighter bounds than the current state-of-the-art bounds obtained via PAC-Bayes with BackProp.

Acknowledgements

The authors would like to acknowledge the use of the University of Oxford Advanced Research Computing (ARC) facility in carrying out this work.⁵ Eugenio Clerico is partly supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the grant EP/R513295/1 (DTP scheme). Arnaud Doucet is partly supported by the EPSRC grant EP/R034710/1. He also acknowledges the support of the UK Defence Science and Technology Laboratory (DSTL) and EPSRC under grant EP/R013616/1. This is part of the collaboration between US DOD, UK MOD, and UK EPSRC, under the Multidisciplinary University Research Initiative.

⁵ <http://dx.doi.org/10.5281/zenodo.22558>

References

- Z. Allen-Zhu, Y. Li, and Y. Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *NeurIPS*, 2019.
- P. Alquier. User-friendly introduction to PAC-Bayes bounds. *arXiv:2110.11216*, 2021.
- P. Alquier and G. Biau. Sparse single-index model. *Journal of Machine Learning Research*, 14, 2013.
- P. Alquier, J. Ridgway, and N. Chopin. On the properties of variational approximations of Gibbs posteriors. *Journal of Machine Learning Research*, 17, 2016.
- A. Ambroladze, E. Parrado-Hernández, and J. Shawe-Taylor. Tighter PAC-Bayes bounds. *NeurIPS*, 2007.
- F. Biggs and B. Guedj. Differentiable PAC-Bayes objectives with partially aggregated neural networks. *Entropy*, 23(10), 2021.
- O. Catoni. PAC-Bayesian supervised classification: The thermodynamics of statistical learning. *IMS Lecture Notes Monograph Series*, 2007.
- E. Clerico, G. Deligiannidis, and A. Doucet. Wide stochastic networks: Gaussian limit and PAC-Bayesian training. *arXiv:2106.09798*, 2021.
- A. S. Dalalyan and A. B. Tsybakov. Sparse regression learning by aggregation and Langevin Monte-Carlo. *Journal of Computer and System Sciences*, 78(5), 2012.
- L. Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6), 2012.
- G. K. Dziugaite and D. M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *UAI*, 2017.
- G. K. Dziugaite and D. M. Roy. Data-dependent PAC-Bayes priors via differential privacy. *NeurIPS*, 2018.
- G. K. Dziugaite, K. Hsu, W. Gharbieh, G. Arpino, and D. M. Roy. On the role of data in PAC-Bayes. *AISTATS*, 2021.
- P. Germain, A. Lacasse, F. Laviolette, and M. Marchand. PAC-Bayesian learning of linear classifiers. *ICML*, 2009.
- B. Guedj. A primer on PAC-Bayesian learning. *Proceedings of the Second Congress of the French Mathematical Society*, 2019.
- Benjamin Guedj and Pierre Alquier. PAC-Bayesian estimation and prediction in sparse additive models. *Electronic Journal of Statistics*, 7, 2013.
- W. Hu, L. Xiao, and J. Pennington. Provable benefit of orthogonal initialization in optimizing deep linear networks. *ICLR*, 2020.
- K. Kawaguchi, L. P. Kaelbling, and Y. Bengio. Generalization in deep learning. *arXiv:1710.05468*, 2017.
- A. Khaled and P. Richtárik. Better theory for SGD in the nonconvex world. *arXiv:2002.03329*, 2020.
- A. Krizhevsky. Learning multiple layers of features from tiny images. *MSc Thesis University of Toronto*, 2009.
- J. Langford and R. Caruana. (Not) bounding the true error. *NeurIPS*, 2002.
- J. Langford and M. Seeger. Bounds for averaging classifiers. *CMU technical report*, 2001.
- J. Langford and J. Shawe-Taylor. PAC-Bayes & margins. *NeurIPS*, 2003.
- G. Letarte, P. Germain, B. Guedj, and F. Laviolette. Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks. *NeurIPS*, 2019.
- A. Maurer. A note on the PAC Bayesian theorem. *arXiv:0411099*, 2004.
- D. A. McAllester. Some PAC-Bayesian theorems. *COLT*, 1998.
- D. A. McAllester. PAC-Bayesian model averaging. *COLT*, 1999.
- D. A. McAllester. PAC-Bayesian stochastic model selection. *Machine Learning*, 51, 2004.
- B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. Exploring generalization in deep learning. *NeurIPS*, 2017.
- E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor, and S. Sun. PAC-Bayes bounds with data dependent priors. *Journal of Machine Learning Research*, 13, 2012.
- T. Poggio, A. Banburski, and Q. Liao. Theoretical issues in deep networks. *PNAS*, 117(48), 2020.
- R. Price. A useful theorem for nonlinear devices having gaussian inputs. *IRE Transactions on Information Theory*, 4(2), 1958.
- M. Pérez-Ortiz, O. Risvaplatá, J. Shawe-Taylor, and C. Szepesvári. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22, 2021a.

M. Pérez-Ortiz, O. Rivasplata, B. Guedj, M. Gleeson, J. Zhang, J. Shawe-Taylor, M. Bober, and J. Kittler. Learning PAC-Bayes priors for probabilistic neural networks. *arXiv:2109.10304*, 2021b.

C. M. Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6), 1981.

V. B. Tadić and A. Doucet. Asymptotic bias of stochastic gradient search. *The Annals of Applied Probability*, 27(6), 2017.

N. Thiemann, C. Igel, O. Wintenberger, and Y. Seldin. A strongly quasiconvex PAC-Bayesian bound. *ALT*, 2017.

C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3), 2021.

W. Zhou, V. Veitch, M. Austern, R. P. Adams, and P. Orbanz. Non-vacuous generalization bounds at the imagenet scale: a PAC-Bayesian compression approach. *ICLR*, 2019.

Supplementary material

SM1 PROOFS

Proposition 2. Denote the cumulative distribution function (CDF) of a standard normal as $\psi : u \mapsto \frac{1}{2}(1 + \operatorname{erf}(u/\sqrt{2}))$. Fix a pair x, y and let

$$L_1 = \psi \left(\max_{i \neq y} \frac{F_i(x) - M_y(x)}{\sqrt{V_y(x)}} \right),$$

$$L_2 = 1 - \prod_{i \neq y} \psi \left(\frac{F_y(x) - M_i(x)}{\sqrt{V_i(x)}} \right),$$

where $F(x) \sim \mathcal{N}(M(x), \operatorname{diag}(V(x)))$. Then

$$\mathbb{E}[L_1] = \mathbb{E}[L_2] = \mathbb{P}(\hat{y} \neq y), \quad (\text{SM1})$$

$$\mathbb{E}[\nabla L_1] = \mathbb{E}[\nabla L_2] = \nabla \mathbb{P}(\hat{y} \neq y), \quad (\text{SM2})$$

where the gradient is with respect to all the components of $M(x)$ and $V(x)$.

Proof. We start by showing that $\mathbb{E}[L_1] = \mathbb{P}(\hat{y} \neq y)$. We have

$$\mathbb{P}(\hat{y} \neq y) = \mathbb{P} \left(F_y(x) < \max_{i \neq y} F_i(x) \right) = \mathbb{E} \left[\mathbb{P} \left(F_y(x) < \max_{i \neq y} F_i(x) \mid \{F_i(x)\}_{i \neq y} \right) \right] = \mathbb{E}[L_1].$$

For L_2 again we first use conditioning w.r.t. $F_y(x)$

$$\mathbb{P}(\hat{y} \neq y) = \mathbb{E} \left[\mathbb{P} \left(F_y(x) < \max_{i \neq y} F_i(x) \mid F_y(x) \right) \right] = 1 - \mathbb{E} \left[\mathbb{P} \left(F_y(x) \geq \max_{i \neq y} F_i(x) \mid F_y(x) \right) \right].$$

As the events $\{F_y(x) \geq F_i(x) \mid F_y(x)\}_{i \neq y}$ are independent, we can write

$$\mathbb{P}(\hat{y} \neq y) = 1 - \mathbb{E} \left[\prod_{i \neq y} \mathbb{P} \left(F_i(x) \leq F_y(x) \mid F_y(x) \right) \right] = \mathbb{E}[L_2],$$

and so (SM1) is proved.

Now, to show (SM2), we need to prove that it is possible to swap expectation and differentiation for both L_1 and L_2 . For L_2 everything is straightforward, as it is a smooth function of M and V (as all the components of V are assumed to be strictly positive) and its gradient can be easily bounded (uniformly in some neighbourhood of $(M_i(x), V_i(x))_{i \neq y}$) by a function of $F_y(x)$ with finite expectation. Hence we can apply Leibniz integral rule. For L_1 , this is the case only for ∂_{M_y} and ∂_{V_y} , as $\max_{i \neq y} \frac{F_i(x) - M_y(x)}{\sqrt{V_y(x)}} = \frac{\max_{i \neq y} \{F_i(x)\} - M_y(x)}{\sqrt{V_y(x)}}$ is smooth in M_y and V_y , and its gradient can be easily bounded (uniformly in some neighbourhood of $(M_y(x), V_y(x))$) by a function of $(F_i(x))_{i \neq y}$ with finite expectation. However, for any $j \neq y$, the integrand is not everywhere differentiable wrt M_j and V_j . Yet, we can still swap expectation and differentiation using Proposition SM1, detailed below. \square

The two results that follow are well known in the literature, and restated here for convenience. For completeness we give a proof for both of them. Denote as $\rho_{m,s}$ the density of a normal random variable with mean m and standard deviation s . For convenience we let $\rho = \rho_{0,1}$. All integrals \int are over \mathbb{R} .

The next proposition is essentially a reformulation of Price's theorem (Price, 1958).

Proposition SM1. Let $Z \sim \mathcal{N}(0, 1)$ and $X = sZ + m$. Let $g : \mathbb{R} \rightarrow \mathbb{R}$ be a locally Lipschitz function with a polynomially bounded derivative. Then

$$\nabla_{m,s} \mathbb{E}_{X \sim \mathcal{N}(m, s^2)}[g(X)] = \mathbb{E}_{Z \sim \mathcal{N}(0, 1)}[\nabla_{m,s} g(sZ + m)].$$

Proof. Recall that $\partial_m \rho_{m,s}(x) = \frac{x-m}{s} \rho_{m,s}(x)$ and $\partial_s \rho_{m,s}(x) = \frac{(x-m)^2 - s^2}{s^3} \rho_{m,s}(x)$. Let $z = sx + m$, then $\rho_{m,s}(x)dx = \rho(z)dz$. Note that by the local Lipschitzianity g' is defined almost everywhere. Since it is polynomially bounded, the expectation $\mathbb{E}[\nabla_{m,s} g(sZ + m)]$ makes sense. Note moreover that g is polynomially bounded as g' is.

We start by proving the equality for the m -derivative. We have

$$\partial_m \mathbb{E}[g(X)] = \partial_m \int \rho_{m,s}(x)g(x)dx = \int (\partial_m \rho_{m,s}(x))g(x)dx,$$

by Leibniz integration rule, as $\rho_{m,s}$ is smooth in its arguments and the continuity and polynomial boundedness of g ensure that $\int x \rightarrow \partial_m \rho_{m,s}(x)g(x)dx$ is well defined and finite. Now, we have

$$\int (\partial_m \rho_{m,s}(x))g(x)dx = \int \frac{x-m}{s^2} \rho_{m,s}(x)g(x)dx = \int \frac{z}{s} \rho(z)g(sz+m)dz.$$

From Lemma SM1 below, we get

$$\int \frac{z}{s} \rho(z)g(sz+m)dz = \int \frac{1}{s} \rho(z)sg'(sz+m)dz = \int \rho(z)g'(sz+m)dz.$$

Now, as $g'(sz+m) = \partial_m g(sz+m)$ we conclude that

$$\partial_m \mathbb{E}[g(X)] = \mathbb{E}[\partial_m g(sZ+m)].$$

For the s -derivative, the proof is essentially analogous. Proceeding as above, we have

$$\partial_s \mathbb{E}[g(X)] = \int (\partial_s \rho_{m,s}(x))g(x)dx = \int \frac{(x-m)^2 - s^2}{s^3} \rho_{m,s}(x)g(x)dx = \int \frac{z^2 - 1}{s} \rho(z)g(sz+m)dz.$$

Again from Lemma SM1 we find that

$$\int \frac{z^2 - 1}{s} \rho(z)g(sz+m)dz = \int \rho(z)zg'(sz+m)dz.$$

We conclude that

$$\partial_s \mathbb{E}[g(x)] = \mathbb{E}[\partial_s g(sz+m)],$$

since $\partial_s g(sz+m) = zg'(sz+m)$. □

The next lemma states Stein's identity (Stein, 1981) and a straightforward corollary.

Lemma SM1. *Let $Z \sim \mathcal{N}(0,1)$, and $g : \mathbb{R} \rightarrow \mathbb{R}$ a locally Lipschitz function with a polynomially bounded derivative. Then*

$$\begin{aligned} \mathbb{E}[Zg(Z)] &= \mathbb{E}[g'(Z)], \\ \mathbb{E}[(Z^2 - 1)g(Z)] &= \mathbb{E}[Zg'(Z)]. \end{aligned}$$

Proof. The first equality, known as Stein's identity, is established using integration by parts:

$$0 = \int (\rho(z)g(z))'dz = \int \rho'(z)g(z)dz + \int \rho(z)g'(z)dz = - \int z\rho(z)g(z) + \int \rho(z)g'(z)dz,$$

where we used that g' exists almost everywhere as g is locally Lipschitz, and that both g and g' are polynomially bounded, so all integral are finite and well defined. Now take $h(z) = zg(z)$. Then we have $h'(z) = zg'(z) + g(z)$ and so

$$\mathbb{E}[Z^2g(Z)] = \mathbb{E}[Zh(Z)] = \mathbb{E}[h'(Z)] = \mathbb{E}[Zg'(Z)] + \mathbb{E}[g(Z)],$$

which is the second equality. □

Proposition 3. *Assume that \mathcal{B} is locally Lipschitz in the hidden stochastic parameters $\theta^{\mathcal{H}}$, and that $\nabla_{\theta^{\mathcal{H}}}\mathcal{B}$ is polynomially bounded. If $\mathcal{B}(\mathcal{E}_S(\mathcal{Q}), \text{Pen})$ is an affine function of $\mathcal{E}_S(\mathcal{Q})$, then we have $\mathbb{E}[\hat{\mathcal{B}}] = \mathcal{B}$ and $\mathbb{E}[\nabla \hat{\mathcal{B}}] = \nabla \mathcal{B}$, the gradient being with respect to the trainable hyper-parameters \mathbf{p} .*

Proof. By linearity it is sufficient to show that $\mathbb{E}[\hat{\mathcal{E}}_S(\mathcal{Q})] = \mathcal{E}_S(\mathcal{Q})$ and $\mathbb{E}[\nabla \hat{\mathcal{E}}_S(\mathcal{Q})] = \nabla \mathcal{E}_S(\mathcal{Q})$. Note that, following the discussion of Section 3.1, we can write $\hat{\mathcal{E}}_S(\mathcal{Q}) = \sum_{x \in S} \hat{\mathcal{E}}_x$ where

$$\hat{\mathcal{E}}_x = E(M(x, \theta^{\mathcal{H}}, \mathbf{p}^{\mathcal{L}}), V(x, \theta^{\mathcal{H}}, \mathbf{p}^{\mathcal{L}}), \xi),$$

for some suitable function E . If we are dealing with binary classification the variable ξ can be omitted, otherwise it represents the random draws needed to obtain the estimate L_1 or L_2 (defined in Proposition 2).

Define $\mathcal{E}_x = \mathbb{E}[\hat{\mathcal{E}}_x]$, the expectation being over ξ and $\theta^{\mathcal{H}}$. By Proposition 2 (if we are dealing with multiclass classification, otherwise by definition) we get that $\mathcal{E}_S(\mathcal{Q}) = \sum_{x \in S} \mathcal{E}_x$. Consequently we have

$$\mathcal{E}_S(\mathcal{Q}) = \sum_{x \in S} \mathbb{E}[\hat{\mathcal{E}}_x] = \mathbb{E}[\hat{\mathcal{E}}_S(\mathcal{Q})].$$

Now, to show the unbiasedness of the gradient, it is enough to show that for all $x \in S$

$$\nabla_{\mathbf{p}} \mathcal{E}_x = \mathbb{E}[\nabla_{\mathbf{p}} \hat{\mathcal{E}}_x].$$

First, again by Proposition 2 we can write

$$\mathbb{E}[\nabla_{\mathbf{p}} \hat{\mathcal{E}}_x] = \mathbb{E}[\nabla_{\mathbf{p}} \mathbb{E}[\hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]] = \mathbb{E}\left[\frac{\partial(M, V)}{\partial \mathbf{p}} \mathbb{E}[\nabla_{M, V} \hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]\right] = \mathbb{E}\left[\frac{\partial(M, V)}{\partial \mathbf{p}} \nabla_{M, V} \mathbb{E}[\hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]\right] = \mathbb{E}[\nabla_{\mathbf{p}} \mathbb{E}[\hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]].$$

Now, $\mathbb{E}[\hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]$ is the probability that a component of a Gaussian vector with mean M and covariance $\text{diag}(V)$ is smaller than the maximum of the other components (cf. Section 3.1). This is a smooth function of M and V , which in turn are smooth functions of the last layer's hyper-parameters $\mathbf{p}^{\mathcal{L}}$. As a consequence we can write

$$\nabla_{\mathbf{p}^{\mathcal{L}}} \mathcal{E}_x = \mathbb{E}[\nabla_{\mathbf{p}^{\mathcal{L}}} \mathbb{E}[\hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]] = \mathbb{E}[\nabla_{\mathbf{p}^{\mathcal{L}}} \hat{\mathcal{E}}_x].$$

As for the hidden hyper-parameters, since we are assuming that all the hidden stochastic parameters are independent Gaussian random variables, we can apply Proposition SM1, which brings

$$\nabla_{\mathbf{p}^{\mathcal{H}}} \mathcal{E}_x = \mathbb{E}[\nabla_{\mathbf{p}^{\mathcal{H}}} \mathbb{E}[\hat{\mathcal{E}}_x | \mathcal{F}^{\mathcal{H}}]] = \mathbb{E}[\nabla_{\mathbf{p}^{\mathcal{H}}} \hat{\mathcal{E}}_x],$$

thus concluding our proof. \square

SM2 A NOTE ON UNBIASEDNESS

The previous results state that the gradient estimates used in the Cond-Gauss algorithm are unbiased, as long as the bound is affine in the empirical error. Under suitable regularity conditions, this ensures that stochastic gradient descent algorithms converge to a stationary point of the objective (Khaled and Richtárik, 2020). However, among the four bounds (4) that we used in our experiments, only (4b) and (4d) are actually affine. We argue here that in most cases of interest $\hat{\mathcal{E}}_S(\mathcal{Q})$ is concentrated enough that the bounds (4a) and (4c) are approximately affine in the empirical error. In the following, we detail this heuristic idea and then give some empirical evidence on MNIST in the case of (4a). This almost affine behaviour ensures that the gradient used by our stochastic optimisation procedure is almost unbiased, and hence we can expect the algorithm to converge to a point close to a stationary point of the objective (Tadić and Doucet, 2017).

Consider a generic bound $\mathcal{B} = B(\mathcal{E}_S(\mathcal{Q}))$, where B might be a non-affine function. Our estimate is of the form $\hat{\mathcal{B}} = B(\hat{\mathcal{E}}_S(\mathcal{Q}))$. We can now consider a linearised version \bar{B} of B , defined as

$$\bar{B}(\mathcal{E}) = B(\mathcal{E}_S(\mathcal{Q})) + (\mathcal{E} - \mathcal{E}_S(\mathcal{Q}))B'(\mathcal{E}_S(\mathcal{Q})).$$

Clearly, in a sufficiently small neighborhood of $\mathcal{E}_S(\mathcal{Q})$, we can expect B and \bar{B} to almost coincide. In particular, if the law of $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates around $\mathcal{E}_S(\mathcal{Q})$, we can expect that with high probability

$$B(\hat{\mathcal{E}}_S(\mathcal{Q})) \simeq \bar{B}(\mathcal{E}_S(\mathcal{Q})).$$

As \bar{B} is affine, we can apply Proposition 3 and get

$$\begin{aligned} \mathbb{E}[\hat{\mathcal{B}}] &= \mathbb{E}[B(\hat{\mathcal{E}}_S(\mathcal{Q}))] \simeq \mathbb{E}[\bar{B}(\hat{\mathcal{E}}_S(\mathcal{Q}))] = \bar{B}(\mathcal{E}_S(\mathcal{Q})) = B(\mathcal{E}_S(\mathcal{Q})) = \mathcal{B}, \\ \mathbb{E}[\nabla_{\mathbf{p}} \hat{\mathcal{B}}] &= \mathbb{E}[\nabla_{\mathbf{p}} B(\hat{\mathcal{E}}_S(\mathcal{Q}))] \simeq \mathbb{E}[\nabla_{\mathbf{p}} \bar{B}(\hat{\mathcal{E}}_S(\mathcal{Q}))] = \nabla_{\mathbf{p}} \bar{B}(\mathcal{E}_S(\mathcal{Q})) = \nabla_{\mathbf{p}} B(\mathcal{E}_S(\mathcal{Q})) = \nabla_{\mathbf{p}} \mathcal{B}. \end{aligned}$$

To empirically justify the above, we consider the bound (4a), which was used for most of our experiments. Figure SM1 and Figure SM2 show that indeed $\hat{\mathcal{E}}_S(\mathcal{Q})$ is sufficiently concentrated around its mean to see the bound as an affine function of the empirical error. Figure SM1 reports the data from the network achieving the best bound in our experiments with *data-dependent* priors on MNIST. On the other hand, among the networks trained with the *invKL* objectives on MNIST with *data-free* priors, the one achieving the tightest bound was used for Figure SM2. In both figures, the histogram represents the distribution of 10000 realisations of $\hat{\mathcal{E}}_S(\mathcal{Q})$. It is clear that in both cases the bound is essentially affine in the empirical loss, in the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates (zoomed portion of the plot).

Similar observations hold when the objective is derived from (4c).

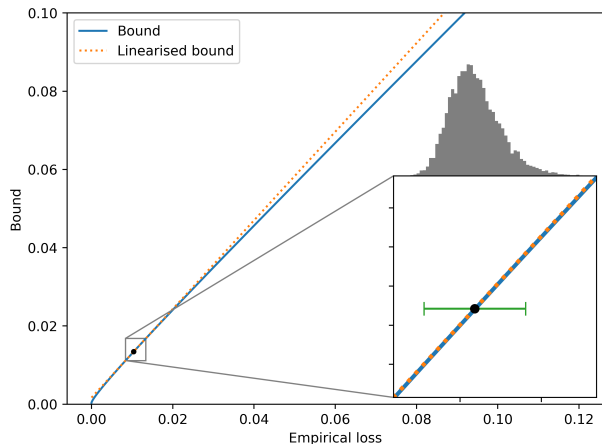


Figure SM1: (Same as Figure 1 from the main text.) Experimental evidence, from a network trained with a *data-dependent* prior on MNIST, that the bound (4a) is almost affine in the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates. The network used was the one achieving the best generalisation bound in our experiment on MNIST with *data-dependent* priors. 10000 realisations of $\hat{\mathcal{E}}_S(\mathcal{Q})$ were sampled. Their distribution is summarised by the histogram above the zoomed portion of the plot. The black dot is the bound for the average value found for $\hat{\mathcal{E}}_S(\mathcal{Q})$, while the green error bar has a total width of 4 empirical standard deviations. In the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates, the bound and its linearised version almost coincide. Along the green error bar, the bound’s slope has a relative variation of $\pm 0.8\%$.

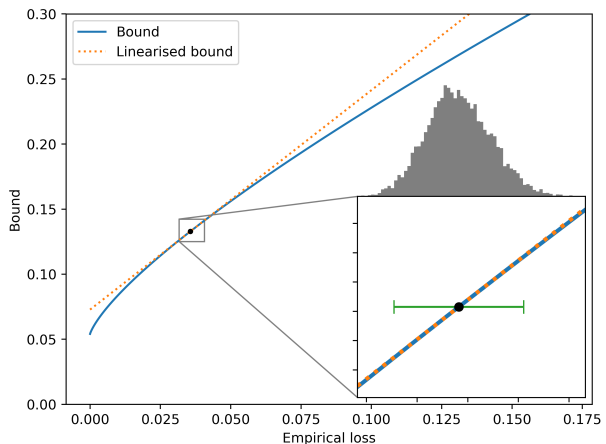


Figure SM2: Experimental evidence, from a network trained with a *data-free* prior on MNIST, that the bound (4a) is almost affine in the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates. Among the networks trained with the *invKL* objectives on MNIST with *data-free* priors, the one achieving the tightest bound was used in this experiment. 10000 realisations of $\hat{\mathcal{E}}_S(\mathcal{Q})$ were sampled. Their distribution is summarised by the histogram above the zoomed portion of the plot. The black dot is the bound for the average value found for $\hat{\mathcal{E}}_S(\mathcal{Q})$, while the green error bar has a total width of 4 empirical standard deviations. In the region where $\hat{\mathcal{E}}_S(\mathcal{Q})$ concentrates, the bound and its linearised version almost coincide. Along the green error bar, the bound’s slope has a relative variation of $\pm 2\%$.

SM3 PAC-BAYESIAN TRAINING FOR GENERAL ARCHITECTURES

In the main text we focused on the case of a network whose stochastic parameters are all Gaussian. This is not a necessary condition for the Cond-Gauss algorithm. What we need is actually to be able to express the KL between prior and posterior as a differentiable expression of the hyper-parameters, and to evaluate the gradient (wrt the hyper-parameters) of a single empirical loss’s realisation. We can satisfy this last requirement if we are able to rewrite the stochastic parameters as a (differentiable) function Θ of the hyper-parameters \mathbf{p} and of some random variable τ (independent of \mathbf{p}) such that $\Theta(\mathbf{p}, \tau)$ has the same law of θ , namely $\mathcal{Q}_{\mathbf{p}}$. In short, for any measurable function φ ,

$$\mathbb{E}_{\theta \sim \mathcal{Q}_{\mathbf{p}}}[\varphi(\theta)] = \mathbb{E}_{\tau}[\varphi(\Theta(\mathbf{p}, \tau))].$$

In particular, to sample a realisation $\hat{\varphi}$ of $\varphi(\theta)$ we can sample a realisation $\hat{\tau}$ of τ and then define

$$\hat{\varphi} = \varphi(\Theta(\mathbf{p}, \hat{\tau})).$$

As long as $\varphi \circ \Theta$ is differentiable in \mathbf{p} , we can evaluate the gradient of $\hat{\varphi}$ wrt \mathbf{p} .

For the Cond-Gauss algorithm to be implementable, we require that there exists a \mathbf{p} -differentiable reparametrisation Θ for the hidden parameters $\theta^{\mathcal{H}}$. Clearly, this is the case if $\theta^{\mathcal{H}}$ is a Gaussian vector with independent components. Indeed, if we denote by $\mathbf{m}^{\mathcal{H}}$ and $\mathbf{s}^{\mathcal{H}}$ the vectors of means and standard deviations, we have

$$\theta^{\mathcal{H}} = \mathbf{m}^{\mathcal{H}} + \mathbf{s}^{\mathcal{H}} \odot \tau,$$

where τ is a vector with independent standard normal components and \odot denotes the component-wise product. This is what was used for the networks in our experiments.

SM4 NUMERICAL EVALUATION OF kl^{-1} AND ITS GRADIENT

When the training objective is `invKL`, it is necessary to evaluate kl^{-1} and its gradient, in order to implement the Cond-Gauss algorithm. Many of the most popular deep learning libraries, such as PyTorch and TensorFlow, do not provide an implementation for kl^{-1} . However, as pointed out by [Dziugaite and Roy \(2017\)](#), a fast numerical evaluation can be done via a few iterations of Newton’s method. This is what we used in our code.

We show here that the gradient of kl^{-1} can be expressed as a function of kl^{-1} , so that the implementation of the latter allows the evaluation of the former. Recall that

$$\text{kl}(u\|v) = u \log \frac{u}{v} + (1-u) \log \frac{1-u}{1-v}.$$

For $u > 0$, the mapping $v \mapsto \text{kl}(u\|v)$ is not injective. However if we restrict its domain to $\{(u, v) \in [0, 1]^2 : v \geq u\}$, then we find a bijective map, whose inverse coincides with $c \mapsto \text{kl}^{-1}(u|c)$ (with the definition (3) for kl^{-1}). It follows immediately that

$$\partial_c \text{kl}^{-1}(u|c) = \frac{1}{\partial_v \text{kl}(u\|v)} \Big|_{v=\text{kl}^{-1}(u|c)} = \left(\frac{1-u}{1-v} - \frac{u}{v} \right)^{-1} \Big|_{v=\text{kl}^{-1}(u|c)}.$$

To find an expression for $\partial_u \text{kl}^{-1}(u|c)$ we can proceed as follow. Let $\text{kl}^{-1}(u|c) = v$ and $\text{kl}^{-1}(u+\varepsilon|c) = v+\varepsilon'$, with $\varepsilon' = \varepsilon \partial_u \text{kl}^{-1}(u|c) + o(\varepsilon)$. This means that $\text{kl}(u+\varepsilon\|v+\varepsilon') = \text{kl}(u\|v)$, so that $\varepsilon \partial_u \text{kl}(u\|v) + \varepsilon' \partial_v \text{kl}(u\|v) = o(\varepsilon)$. Taking $\varepsilon \rightarrow 0$ we find

$$\partial_u \text{kl}^{-1}(u|c) = - \frac{\partial_u \text{kl}(u\|v)}{\partial_v \text{kl}(u\|v)} \Big|_{v=\text{kl}^{-1}(u|c)} = \left(\log \frac{1-u}{1-v} - \log \frac{u}{v} \right) / \left(\frac{1-u}{1-v} - \frac{u}{v} \right) \Big|_{v=\text{kl}^{-1}(u|c)}.$$

SM5 ADDITIONAL EXPERIMENTAL DETAILS AND RESULTS

In this section we give additional details about our experiments. The PyTorch code written for this paper is available at <https://github.com/eclerico/CondGauss>. In all our experiments we used the average of 100 independent estimates of L^1 (defined in Proposition 2) to evaluate the empirical error. To keep the standard deviations σ positive during the training, we trained the parameters ρ defined by $\sigma = |\rho|^{3/2}$. We found empirically that this transformation allowed for a much faster training compared to the usual exponential choices ([Dziugaite and Roy, 2017](#); [Pérez-Ortiz et al., 2021a](#)).

SM5.1 MNIST

For our experiments on MNIST, we only used the standard training dataset, which consists of 60000 labelled examples. We ran our experiments on a 4-layer ReLU stochastic network, whose parameters were independent Gaussians with trainable means and variances. The architecture used was the following:

$$x \mapsto y = L_2 \circ \phi \circ L_1 \circ \phi \circ f \circ C_2 \circ \phi \circ C_1(x),$$

with

- C_1 : convolutional layer; channels: IN 1, OUT 32; kernel: (3, 3); stride: (1, 1);
- C_2 : convolutional layer; channels: IN 32, OUT 64; kernel: (3, 3); stride: (1, 1);
- L_1 : linear layer; dimensions: IN 9216, OUT 128;

- L_2 : linear layer; dimensions: IN 128, OUT 10;
- f : max pool (kernel size = 2) & flatten;
- ϕ : ReLU activation component-wise.

All convolutional and linear layers were with bias.

SM5.1.1 Data-free priors

We first experimented on data-free priors, whose means were initialised via the Pytorch default initialisation. We tried different values for the initial prior’s variances: .01, .001, and .0001. We compared the performances of the standard PAC-Bayesian training algorithm (S), where the misclassification loss is replaced by a bounded version of the cross-entropy loss as in Pérez-Ortiz et al. (2021a), and the Cond-Gauss algorithm (G). We used the following four training objectives from (4):

$$\begin{aligned}
 \text{invKL} &: && \text{kl}^{-1}(\mathcal{E}_S(\mathcal{Q})|\text{Pen}_\kappa); \\
 \text{McAll} &: && \mathcal{E}_S(\mathcal{Q}) + \sqrt{\text{Pen}_\kappa/2}; \\
 \text{quad} &: && (\sqrt{\mathcal{E}_S(\mathcal{Q}) + \text{Pen}_\kappa/2} + \sqrt{\text{Pen}_\kappa/2})^2; \\
 \text{lbd} &: && \frac{1}{1-\lambda/2}(\mathcal{E}_S(\mathcal{Q}) + \text{Pen}_\kappa/\lambda),
 \end{aligned}$$

where the KL penalty is defined as

$$\text{Pen}_\kappa = \frac{\kappa}{m} \left(\text{KL}(\mathcal{Q}||\mathcal{P}) + \log \frac{2\sqrt{m}}{\delta} \right). \tag{9}$$

The factor κ in (9) can increase or reduce the weight of the KL term during the training. We experimented three different values for this parameter: 0.5, 1, and 2. For the last objective, `lbd`, the parameter λ takes values in (0, 1) and is optimised during training⁶.

For all the different training settings, the network was trained via SGD with momentum for 250 epochs with a learning rate $\eta = .005$ followed by 50 epochs with $\eta = .0001$. We tried using different values for the momentum: 0.5, 0.7, and 0.9. During the training, at the end of each epoch, we kept track of the bound (4a)’s empirical value in order to pick the best epoch at the end of the training.

Figure 2 and Table 1 in the main text report our results.

SM5.1.2 Data-dependent priors

For the data-dependent priors, we used 50% of the dataset to train \mathcal{P} and the remaining 50% to train \mathcal{Q} . We always used the Cond-Gaussian algorithm for both prior and posterior. All the posteriors were trained with the `invKL` objective and $\kappa = 1$, whilst for the prior, we experimented with both `invKL` (with $\kappa = 0.1$) and with direct empirical risk minimisation (ERM), meaning that the objective was simply $\mathcal{E}_S(\mathcal{Q})$. The initial prior’s variances were set at 0.01, while the means were randomly initialised (via the default PyTorch initialisation for each layer). We used different dropout values, as shown in Table SM1. The prior’s training consisted of 750 epochs with $\eta = .005$, followed by 250 epochs with $\eta = .0001$, the posterior’s training of 750 epochs with $\eta = 10^{-5}$, followed by 250 epochs with $\eta = 10^{-6}$. We used SGD with a momentum of 0.9 for both priors and posteriors. The results of the experiment can be found in Table SM1.

SM5.2 CIFAR10

As we had done for the MNIST dataset, for CIFAR10 we used only the standard training dataset (50000 labelled images). We trained a 9-layer architecture (6 convolutional + 3 linear layers) and a 15-layer architecture (12 convolutional + 3 linear layers). We experimented with data-dependent priors only, training \mathcal{P} with 50% of the data for the 9-layer classifier and both with 50% and 70% for the 15-layer one.

⁶In our experiments, we initialised λ at 0.5 and then doubled the number of epochs, alternating one epoch of λ ’s optimisation with one of optimisation for m and \mathfrak{s} .

Table SM1: MNIST - Prior learnt on 50% of the dataset

Prior							Posterior					
tm ^a	do ^b	pf ^c	iv ^d	l1 ^e	l2 ^f	p ^g	tm ^a	l1 ^e	l2 ^f	t ^h	p ^g	b ⁱ
G ERM	0	-	.001	.0010	.0126	3.179	G invKL	.0010	.0122	.0122 \pm .0006	3.671	.0164
G invKL	0	.01	.001	.0008	.0125	3.179	G invKL	.0008	.0119	.0115 \pm .0007	3.882	.0162
G ERM	.1	-	.001	.0010	.0111	3.179	G invKL	.0010	.0107	.0110 \pm .0006	3.688	.0148
G invKL	.1	.01	.001	.0006	.0113	3.179	G invKL	.0006	.0107	.0109 \pm .0006	3.944	.0149
G ERM	.2	-	.001	.0011	.0111	3.179	G invKL	.0011	.0107	.0101 \pm .0005	3.742	.0148
G invKL	.2	.01	.001	.0010	.0108	3.179	G invKL	.0010	.0104	.0101 \pm .0006	3.801	.0144

^a tm: Training method.

^b do: Dropout probability for the prior’s training.

^c pf: Penalty factor κ for the prior’s training objective.

^d iv: Initial value of the prior’s variances.

^e l1: Empirical error estimate on the prior dataset.

^f l2: Empirical error estimate on the posterior dataset.

^g p: KL penalty **Pen (2)** in 10^{-4} units.

^h t: Test error \pm standard deviation (from 1000 realisations).

ⁱ b: Final PAC-Bayesian bound.

SM5.2.1 9-layer architecture

The 9-layer architecture had the following structure:

$$x \mapsto L_3 \circ \phi \circ L_2 \circ \phi \circ L_1 \circ \phi \circ f_2 \circ C_6 \circ \phi \circ C_5 \circ \phi \circ f_1 \circ C_4 \circ \phi \circ C_3 \circ \phi \circ f_1 \circ C_2 \circ \phi \circ C_1(x).$$

Here are detailed the different layers:

- C_1 : convolutional layer; channels: IN 3, OUT 32; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_2 : convolutional layer; channels: IN 32, OUT 64; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_3 : convolutional layer; channels: IN 64, OUT 128; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_4 : convolutional layer; channels: IN 128, OUT 128; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_5 : convolutional layer; channels: IN 128, OUT 256; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_6 : convolutional layer; channels: IN 256, OUT 256; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- L_1 : linear layer; dimensions: IN 4096, OUT 1024;
- L_2 : linear layer; dimensions: IN 1024, OUT 512;
- L_3 : linear layer; dimensions: IN 512, OUT 10;
- f_1 : max pool (kernel size = 2, stride = 2);
- f_2 : max pool (kernel size = 2, stride = 2) & flatten;
- ϕ : ReLU activation component-wise.

All convolutional and linear layers are with bias.

The results for the 9-layer architecture are reported in Table 2 in the main text. After some preliminary experiments, we chose to train both priors and posteriors via the Cond-Gauss algorithm with the invKL objective. We used a small factor κ for the prior, to avoid regularising too much, whilst κ was 1 for the posterior. We tried different values for the dropout and κ in the prior’s training (see Table 2). We used SGD with momentum 0.9 for both prior and posterior. For \mathcal{P} the training consisted of 1500 epochs with $\eta = .005$ followed by 500 epochs with $\eta = .0001$, whilst \mathcal{Q} was trained for 1500 epochs with $\eta = 10^{-5}$, plus 500 epochs with $\eta = 10^{-6}$.

SM5.2.2 15-layer architecture

The 15-layer architecture had the following structure:

$$x \mapsto L_3 \circ \phi \circ L_2 \circ \phi \circ L_1 \circ \phi \circ f_2 \circ C_{12} \circ \phi \circ C_{11} \circ \phi \circ C_{10} \circ \phi \circ C_9 \circ \phi \circ f_1 \circ C_8 \circ \phi \circ C_7 \circ \phi \circ f_2 \circ C_6 \circ \phi \circ C_5 \circ \phi \circ f_1 \circ C_4 \circ \phi \circ C_3 \circ \phi \circ f_1 \circ C_2 \circ \phi \circ C_1(x).$$

Here are detailed the different layers:

- C_1 : convolutional layer; channels: IN 3, OUT 32; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_2 : convolutional layer; channels: IN 32, OUT 64; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_3 : convolutional layer; channels: IN 64, OUT 128; kernel: (3, 3); stride: (1, 1); padding(1, 1);

Table SM2: CIFAR10 - 15 layers - Prior learnt on 50% and 70% of the dataset

		Prior						Posterior						
		tm ^a	do ^b	pf ^c	iv ^d	l1 ^e	l2 ^f	p ^g	tm ^a	l1 ^e	l2 ^f	t ^h	p ^g	b ⁱ
Prior trained on 50% of the dataset														
Pre-Train do=.1	G ERM	0	-	.001	.0090	.1946	3.778	G invKL	.0090	.1924	.1933 \pm .0020	4.775	.2082	
	G invKL	0	.01	.001	.0085	.1937	3.778	G invKL	.0084	.1909	.1922 \pm .0022	4.913	.2068	
	G ERM	.1	-	.001	.0139	.1722	3.778	G invKL	.0139	.1709	.1736 \pm .0018	4.386	.1855	
	G invKL	.1	.01	.001	.0222	.1746	3.778	G invKL	.0222	.1725	.1760 \pm .0020	4.703	.1875	
Pre-Train do=.2	G ERM	0	-	.001	.0214	.1996	3.778	G invKL	.0214	.1974	.1939 \pm .0020	4.734	.2133	
	G invKL	0	.01	.001	.0169	.1963	3.778	G invKL	.0169	.1941	.1930 \pm .0022	4.859	.2100	
	G ERM	.1	-	.001	.0240	.1772	3.778	G invKL	.0240	.1758	.1791 \pm .0017	4.474	.1907	
	G invKL	.1	.01	.001	.0394	.1764	3.778	G invKL	.0393	.1747	.1734 \pm .0019	4.606	.1897	
Prior trained on 70% of the dataset														
Pre-Train do=.1	G ERM	0	-	.001	.0057	.1616	6.127	G invKL	.0057	.1602	.1643 \pm .0020	6.882	.1774	
	G invKL	0	.01	.001	.0062	.1634	6.127	G invKL	.0062	.1617	.1648 \pm .0021	7.203	.1793	
	G ERM	.1	-	.001	.0098	.1443	6.127	G invKL	.0098	.1430	.1470 \pm .0017	7.006	.1595	
	G invKL	.1	.01	.001	.0180	.1467	6.127	G invKL	.0178	.1446	.1506 \pm .0019	7.374	.1616	
Pre-Train do=.2	G ERM	0	-	.001	.0151	.1639	6.127	G invKL	.0151	.1622	.1696 \pm .0018	7.161	.1797	
	G invKL	0	.01	.001	.0127	.1629	6.127	G invKL	.0127	.1611	.1656 \pm .0020	7.293	.1787	
	G ERM	.1	-	.001	.0175	.1484	6.127	G invKL	.0175	.1471	.1506 \pm .0016	7.043	.1638	
	G invKL	.1	.01	.001	.0306	.1500	6.127	G invKL	.0305	.1484	.1498 \pm .0018	7.090	.1652	

^a tm: Training method.^b do: Dropout probability for the prior’s training.^c pf: Penalty factor κ for the prior’s training objective.^d iv: Initial value of the prior’s variances.^e l1: Empirical error estimate on the prior dataset.^f l2: Empirical error estimate on the posterior dataset.^g p: KL penalty Pen (2) in 10^{-4} units.^h t: Test error \pm standard deviation (from 1000 realisations).ⁱ b: Final PAC-Bayesian bound.

- C_4 : convolutional layer; channels: IN 128, OUT 128; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_5 : convolutional layer; channels: IN 128, OUT 256; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_6 : convolutional layer; channels: IN 256, OUT 256; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_7 : convolutional layer; channels: IN 256, OUT 256; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_8 : convolutional layer; channels: IN 256, OUT 256; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_9 : convolutional layer; channels: IN 256, OUT 512; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_{10} : convolutional layer; channels: IN 512, OUT 512; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_{11} : convolutional layer; channels: IN 512, OUT 512; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- C_{12} : convolutional layer; channels: IN 512, OUT 512; kernel: (3, 3); stride: (1, 1); padding(1, 1);
- L_1 : linear layer; dimensions: IN 2048, OUT 1024;
- L_2 : linear layer; dimensions: IN 1024, OUT 512;
- L_3 : linear layer; dimensions: IN 512, OUT 10;
- f_1 : max pool (kernel size = 2, stride = 2);
- f_2 : max pool (kernel size = 2, stride = 2) & flatten;
- ϕ : ReLU activation component-wise.

All convolutional and linear layers are with bias.

For the 15-layer architecture, we experimented different prior trainings, with 50% and 70% of the training dataset. In both cases, it was necessary to introduce an initial pre-training for the prior’s means, as otherwise the Cond-Gauss algorithm alone could not significantly decrease the training objective. First, we initialised the means with an orthogonal initialisation, as suggested in [Hu et al. \(2020\)](#). Then we optimised them by training a deterministic network (with the same architecture) using the cross-entropy loss on the prior’s dataset, for 50 epochs with $\eta = .005$. Finally, via the Cond-Gauss algorithm, we completed the prior’s training and proceeded with the posterior’s tuning following the same learning rate schedule as for the 9-layer case. We always used SGD with momentum 0.9. Different objectives and dropout factors were used for training the prior, as detailed in Table SM2, which also reports the results of our experiment.

3.3 Statements of authorship

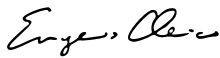
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

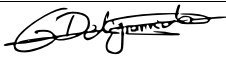
Title of Paper	Wide stochastic networks: Gaussian limit and PAC-Bayesian training
Publication Status	Published
Publication Details	E. Clerico, G. Deligiannidis, and A. Doucet. Wide stochastic networks: Gaussian limit and PAC-Bayesian training. ALT, 2023.

Student Confirmation

Student Name:	Eugenio Clerico		
Contribution to the Paper	I am the first author of this paper. I came out with the main idea and worked on theory and experimental results. George Deligiannidis and Arnaud Doucet provided useful insights and helped with the writing of the paper and the checking of the proofs.		
Signature		Date	24/03/2023

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:	Prof George Deligiannidis		
Supervisor comments	Eugenio's description of his contributions to the paper is fair and accurate		
Signature		Date	27/03/2023

This completed form should be included in the thesis, at the end of the relevant chapter.

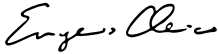
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).


Title of Paper	Conditionally Gaussian PAC-Bayes
Publication Status	Published
Publication Details	E. Clerico, G. Deligiannidis, and A. Doucet. Conditionally Gaussian PAC-Bayes. AISTATS, 2022.

Student Confirmation

Student Name:	Eugenio Clerico		
Contribution to the Paper	I am the first author of this paper. I came out with the main idea and worked on theory and experimental results. George Deligiannidis and Arnaud Doucet provided useful insights and helped with the writing of the paper and the checking of the proofs.		
Signature		Date	24/03/2023

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:	Prof George Deligiannidis		
Supervisor comments	Eugenio's description of his contributions to the paper is fair and accurate		
Signature		Date	27/03/2023

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 4

Chained generalisation bounds

Chained Generalisation Bounds

Eugenio Clerico

Amitis Shidani

George Deligiannidis

Arnaud Doucet

Department of Statistics, University of Oxford, UK.

CLERICO@STATS.OX.AC.UK

SHIDANI@STATS.OX.AC.UK

DELIGIAN@STATS.OX.AC.UK

DOUCET@STATS.OX.AC.UK

Editors: Po-Ling Loh and Maxim Raginsky

Abstract

This work discusses how to derive upper bounds for the expected generalisation error of supervised learning algorithms by means of the chaining technique. By developing a general theoretical framework, we establish a duality between generalisation bounds based on the regularity of the loss function, and their chained counterparts, which can be obtained by lifting the regularity assumption from the loss onto its gradient. This allows us to re-derive the chaining mutual information bound from the literature, and to obtain novel chained information-theoretic generalisation bounds, based on the Wasserstein distance and other probability metrics. We show on some toy examples that the chained generalisation bound can be significantly tighter than its standard counterpart, particularly when the distribution of the hypotheses selected by the algorithm is very concentrated.

Keywords: Generalisation bounds; Chaining; Information-theoretic bounds; Mutual information; Wasserstein distance; PAC-Bayes.

1. Introduction

In the supervised setting, a learning algorithm is a procedure that takes a training dataset as input and returns a hypothesis (*e.g.*, regression coefficients, weights of a neural network, etc.). Ideally, the learned hypothesis should perform well on both the input dataset and new data, which were not used for the training. There is hence interest in providing generalisation bounds, namely upper bounds on the algorithm’s gap in performance for seen and unseen instances.

The first generalisation bounds were based on characterisations of the hypothesis space’s complexity, such as the VC dimension or the Rademacher complexity (Bousquet et al., 2004; Vapnik, 2000; Shalev-Shwartz and Ben-David, 2014). However, due to their algorithm-independent nature, these bounds must hold even for the worst algorithm on the given hypothesis space. Consequently, they are often inadequate for modern over-parameterised neural networks, with the complexity measure usually scaling exponentially with the architecture’s depth (Anthony and Bartlett, 2002; Zhang et al., 2021; Belkin et al., 2018).

To address this issue, recent approaches aim at providing algorithm-dependent generalisation bounds. The underlying intuition is that if the output hypothesis is less dependent on the input dataset, it would be less prone to overfitting, and so generalises better. Among the results building on this idea, there are bounds based on uniform stability (Bousquet and Elisseeff, 2002) and differential privacy (Dwork and Roth, 2014), PAC-Bayesian bounds (Guedj, 2019; McAllester, 1998, 1999), and information-theoretic bounds.

In this paper, we shall mainly focus on the information-theoretic framework, where the learning algorithm is seen as a noisy channel connecting the input dataset and the chosen hypothesis.

Russo and Zou (2019) and Xu and Raginsky (2017) were the first to introduce this approach. They upper-bounded the expected generalisation error via the Mutual Information (MI) between the input sample and the learnt hypothesis. This bound is simple and can be applied to a broad class of learning algorithms. However, a major drawback is that it becomes infinite if the choice of the hypothesis is deterministic in the input. Motivated by this problem, several strategies have been proposed.

Bu et al. (2019) gave an individual-sample MI bound, while Steinke and Zakyntinou (2020) introduced a conditional version of the MI, which is always finite. Rodríguez-Gálvez et al. (2020), Haghifam et al. (2020), and Hellström and Durisi (2020) extended and merged these results. Alternatively, different measures of algorithmic stability can replace the MI: Lopez and Jog (2018), Wang et al. (2019), and Rodríguez-Gálvez et al. (2021) proposed bounds based on the Wasserstein distance, while others focused on total variation, f -divergences, and lautum information (Wang et al., 2019; Rodríguez-Gálvez et al., 2021; Esposito et al., 2021; Palomar and Verdú, 2008).

Adopting a different perspective, Asadi et al. (2018) observed that several information-theoretic bounds fail to exploit the dependencies between hypotheses. They hence proposed to combine the original MI bound with the chaining method, a powerful tool from high dimensional probability originally aimed at upper-bounding the expected supremum of random processes. First introduced by Kolmogorov (see van Handel (2016)), the chaining technique has been successfully extended and developed (Dudley, 1967; Talagrand, 2005, 2014). In their Chaining Mutual Information (CMI) bound, Asadi et al. (2018) take finer and finer discretisations of the hypothesis space and rewrite the generalisation error as a telescopic sum, whose terms can be controlled by exploiting the dependencies between the hypotheses. Subsequently, Asadi and Abbe (2020) adapted the CMI technique to the architecture of deep neural nets, while Zhou et al. (2022) introduced bounds based on a stochastic version of chaining. However, it is worth mentioning that previous works had already applied the chaining method to algorithm-dependent bounds. For instance, Audibert and Bousquet (2004) combined the generic chaining from Talagrand (2005) with the PAC-Bayesian approach.

As a final comment, it must be noted that the generalisation bounds from the information-theoretic literature are hard to evaluate in practice, involving expectations with respect to the unknown sample distribution. Nevertheless, they provide useful intuition on the mechanism of the learning process and, as a result, they represent a very active research area. Moreover, recent works have built on them to derive computable analytical bounds for specific algorithms, such as Langevin dynamics, stochastic gradient Langevin dynamics, and stochastic gradient descent (Bu et al., 2019; Negrea et al., 2019; Haghifam et al., 2020; Rodríguez-Gálvez et al., 2020; Neu et al., 2021).

1.1. Our contributions

The CMI bound is an interesting multi-scale reformulation of the original MI result by Russo and Zou (2019). However, in the information-theoretic literature on generalisation bounds, the chaining method has been coupled only with the MI (Asadi et al., 2018; Asadi and Abbe, 2020; Zhou et al., 2022). Two questions then naturally arise. *Is it possible to derive chained versions of other kinds of generalisation bounds? Can these chained bounds be tighter than their original counterparts?*

In the present work, we establish a duality that reads as follows. *Each bound, based on (a certain notion of) regularity of the loss function, corresponds to a chained bound that can be obtained by lifting the regularity condition from the loss to its gradient.* To make sense of this, we first introduce a general framework, standardising the main step in the proof of several information-theoretic bounds from the literature. We then discuss how to extend this framework leveraging the

chaining technique, and we provide a simple method to derive novel chained generalisation bounds. We show indeed that in our framework each unchained bound corresponds to a chained one (see Theorems 2 and 4), in a way reflecting the connection between the MI and CMI results.

The framework introduced in this work encompasses several information-theoretic *backward-channel*¹ bounds, and allows us to derive their chained counterparts. However, due to space limitations, many explicit results are deferred to Appendix G (see Table 1) and in the main text we focus on four bounds to concretely illustrate how our framework works: the MI bound from Russo and Zou (2019) and the CMI bound from Asadi et al. (2018) serve as a motivation for our general result, while as an application of our framework we derive a novel Wasserstein bound (see Proposition 15), which is the chained counterpart of a bound from Lopez and Jog (2018).

Moreover, we discuss some possible extensions of our work. On the one hand, our information-theoretic framework can be restated with weaker regularity assumptions on both the loss and the hypothesis space. On the other hand, we present an additional bound that does not fit our theoretical framework but can still be derived using essentially the same technical machinery. It is a chained PAC-Bayesian generalisation result, which has the interesting features of being finite even for deterministic algorithms and not requiring the loss to be bounded by a small constant.

As a final remark, there is no generic answer on whether the chained bounds are tighter than their unchained counterparts. However, the chaining technique turns out to be particularly effective when the hypotheses' distribution is very concentrated. In fact, many of the standard bounds do not exploit this feature, the most pathological case being the MI bound, which can even be infinite. In contrast, the chained bounds can be significantly tighter, intrinsically leveraging the dependencies between different hypotheses. We illustrate this phenomenon through some simple toy examples.

2. Preliminaries

Let the input space $(\mathcal{X}, d_{\mathcal{X}})$ be a separable complete metric space, and $\Sigma_{\mathcal{X}}$ the corresponding Borel σ -algebra. We define $\mathcal{S} = \mathcal{X}^m$ and consider a metric $d_{\mathcal{S}}$ inducing the product σ -algebra $\Sigma_{\mathcal{S}} = \Sigma_{\mathcal{X}}^{\otimes m}$. We denote the training dataset as $s = \{x_1, \dots, x_m\} \in \mathcal{S}$. Let \mathbb{P}_X be a probability measure on \mathcal{X} and X a random variable with law \mathbb{P}_X . $S = \{X_1, \dots, X_m\} \in \mathcal{S}$ denotes the random training sample, with law \mathbb{P}_S . We will always assume that the marginal $\mathbb{P}_{X_i} = \mathbb{P}_X$, for each index i . This is of course the case if the X_i are i.i.d. ($\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$). We will suppose that the hypothesis space \mathcal{W} is a closed subset of \mathbb{R}^d , endowed with its Borel σ -algebra $\Sigma_{\mathcal{W}}$. A learning algorithm consists in a Markov kernel that maps each $s \in \mathcal{S}$ to a probability measure $\mathbb{P}_{W|S=s}$ on \mathcal{W} . In turn, this defines a joint probability $\mathbb{P}_{W,S}$ on $\mathcal{W} \times \mathcal{S}$. We denote as \mathbb{P}_W and \mathbb{P}_S the marginal distributions of $\mathbb{P}_{W,S}$, and we let $s \mapsto \mathbb{P}_{W|S=s}$ and $w \mapsto \mathbb{P}_{S|W=w}$ be regular conditional probabilities².

In the supervised framework, the goal is to approximate a map $x \mapsto f_{\star}(x)$ by making use of the information contained in the training sample s (the value of $f_{\star}(x_i)$ is known for each $x_i \in s$). Each hypothesis w represents a parameterised mapping $x \mapsto f_w(x)$, and the training process consists in tuning w , so as to approximate f_{\star} . The loss $\ell : \mathcal{W} \times \mathcal{X} \rightarrow \mathbb{R}$, allows to assess how far each $f_w(x)$ is

1. In the information-theoretic literature, the *forward-channel* connects the sample to the hypothesis, while the *backward-channel* goes the other way. Chaining on the hypotheses combines naturally with the *backward-channel*.

2. The existence of these is ensured by the fact that \mathcal{S} and \mathcal{W} are Polish spaces, cf. Theorem 10.2.2 in Dudley (2002).

from $f^*(x)$. We will always assume that $\ell(w, \cdot) \in L^1(\mathbb{P}_X)$. Define the empirical and the true loss

$$\mathcal{L}_s(w) = \frac{1}{m} \sum_{i=1}^m \ell(w, x_i); \quad \mathcal{L}_{\mathcal{X}}(w) = \mathbb{E}_{\mathbb{P}_X}[\ell(w, X)].$$

We call generalisation error the difference $g_s(w) = \mathcal{L}_{\mathcal{X}}(w) - \mathcal{L}_s(w)$. In this work, we are essentially interested in upper-bounding its expected value $\mathcal{G} = \mathbb{E}_{\mathbb{P}_{W,S}}[g_S(W)]$.

The equality $\mathbb{E}_{\mathbb{P}_{W,S}}[\mathcal{L}_{\mathcal{X}}(W)] = \mathbb{E}_{\mathbb{P}_{W \otimes S}}[\mathcal{L}_S(W)]$, where $\mathbb{P}_{W \otimes S} = \mathbb{P}_W \otimes \mathbb{P}_S$, follows from $\mathcal{L}_{\mathcal{X}}(w) = \mathbb{E}_{\mathbb{P}_S}[\mathcal{L}_S(w)]$ and is the starting point of several information-theoretic bounds. Indeed,

$$\mathcal{G} = \mathbb{E}_{\mathbb{P}_{W \otimes S}}[\mathcal{L}_S(W)] - \mathbb{E}_{\mathbb{P}_{W,S}}[\mathcal{L}_S(W)]$$

can be upper-bounded in terms of how ‘‘far apart’’ $\mathbb{P}_{W,S}$ and $\mathbb{P}_{W \otimes S}$ are.

2.1. Further notation and conventions

The following notation will be used throughout the rest of the paper. $(\mathcal{Z}, \Sigma_{\mathcal{Z}})$ denotes a generic separable complete metric space, endowed with the Borel σ -algebra induced by its metric $d_{\mathcal{Z}}$. We endow \mathcal{P} , the space of all the probability measures on $(\mathcal{Z}, \Sigma_{\mathcal{Z}})$, with the topology of the weak convergence, and we denote the corresponding Borel σ -algebra as $\Sigma_{\mathcal{P}}$. For two coupled random variables Z, Z' on \mathcal{Z} , we write $\mathbb{P}_{Z \otimes Z'}$ for the independent coupling $\mathbb{P}_Z \otimes \mathbb{P}_{Z'}$. For $v, v' \in \mathbb{R}^q$ (for a generic $q \in \mathbb{N}$) we write $\|v\|$ and $v \cdot v'$ for the Euclidean norm and the standard dot product in \mathbb{R}^q respectively. For a random vector $V \in \mathbb{R}^q$, we write that $V \in L^1$ if $\mathbb{E}_{\mathbb{P}_V}[\|V\|] < +\infty$. When we need to specify the integrability of V with respect to a particular law μ , we explicitly write $V \in L^1(\mu)$, that is $\mathbb{E}_{\mu}[\|V\|] < +\infty$. Finally, ξ denotes an arbitrary non-negative real number.

3. General framework

3.1. Bounds based on the regularity of the loss

Both the standard MI and Wasserstein bounds from [Russo and Zou \(2019\)](#) and [Lopez and Jog \(2018\)](#) (see Propositions 10 and 11 in Section 4 for the explicit statements) build on some regularity condition on the dependence of ℓ in x , holding uniformly on \mathcal{W} . As this is a common assumption for various *backward-channel* bounds in the literature, we will now introduce a unified abstract framework, which allows us to re-derive several information-theoretic bounds, such as many of those based on MI, Wasserstein distances, and other probability metrics. Due to the limited space, in the main text we only give a few concrete applications of our framework (see Section 4). A wide range of additional explicit examples, listed in Table 1, can be found in Appendix G.

Definition 1 (\mathfrak{D} -regularity) *Let \mathfrak{D} be a measurable³ map $\mathcal{P} \times \mathcal{P} \rightarrow [0, +\infty]$. Fix $\mu \in \mathcal{P}$ and $\xi \geq 0$. We say that $f : \mathcal{Z} \rightarrow \mathbb{R}$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$, with respect to μ , if $f \in L^1(\mu)$ and, for every $\nu \in \mathcal{P}$ such that $\text{Supp}(\nu) \subseteq \text{Supp}(\mu)$ and $f \in L^1(\nu)$,*

$$|\mathbb{E}_{\mu}[f(Z)] - \mathbb{E}_{\nu}[f(Z)]| \leq \xi \mathfrak{D}(\mu, \nu).$$

We can extend the definition to functions taking values in \mathbb{R}^q , for $q > 1$. We say that $F : \mathcal{Z} \rightarrow \mathbb{R}^q$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ (wrt μ) if $z \mapsto v \cdot F(z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi\|v\|)$ (wrt μ), for all $v \in \mathbb{R}^q$.

3. The measurability wrt $\Sigma_{\mathcal{P}}$ is a technical assumption that is required in order to ensure that expressions, such as $\int_{\mathcal{W}} \mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W=w}) d\mathbb{P}_W(w)$ in Theorem 2, make sense. The reader can be assured that it holds whenever \mathfrak{D} is a measurable function of an f -divergence, or the Wasserstein distance. We refer to Appendix F for more details.

The concept of \mathfrak{D} -regularity is intrinsically connected to the choice of the measure $\mu \in \mathcal{P}$, in the sense that f might be $\mathcal{R}_{\mathfrak{D}}(\xi)$ regular with respect to μ , but not with respect to some other $\mu' \in \mathcal{P}$. For two simple concrete examples of \mathfrak{D} -regularity, we refer to Lemma 9, in Section 4.

Now, let $\mathcal{Z} = \mathcal{S}$ and recall that \mathcal{W} is a closed subset of \mathbb{R}^d , with Borel σ -algebra $\Sigma_{\mathcal{W}}$. On the product space $(\mathcal{W} \times \mathcal{S}, \Sigma_{\mathcal{W}} \otimes \Sigma_{\mathcal{S}})$, we consider a probability measure $\mathbb{P}_{W,S}$, with marginals \mathbb{P}_W and \mathbb{P}_S . Recall that since \mathcal{S} is a Polish space, $w \mapsto \mathbb{P}_{S|W=w}$ defines a regular conditional probability (cf. Theorem 10.2.2 in Dudley (2002)). The next result, which follows easily from the definition of regularity, is a powerful tool to derive generalisation bounds.

Theorem 2 *Assume that $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt \mathbb{P}_S , $\forall w \in \mathcal{W}$. Then we have*

$$|\mathcal{G}| = |\mathbb{E}_{\mathbb{P}_{W \otimes S}}[\mathcal{L}_S(W)] - \mathbb{E}_{\mathbb{P}_{W,S}}[\mathcal{L}_S(W)]| \leq \xi \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W})],$$

where $\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W})] = \int_{\mathcal{W}} \mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W=w}) d\mathbb{P}_W(w)$.⁴

By specialising the concept of \mathfrak{D} -regularity, we can leverage the framework introduced so far and obtain generalisation bounds based on various probability divergences (cf. Table 1). Moreover, individual-sample bounds such as those from Bu et al. (2019) can fit in our framework, as well as bounds based on the random sub-sampling from a super-sample, in the same spirit of the conditional MI bound from Steinke and Zakythinou (2020). We refer the reader to Appendix G for a more detailed discussion of these results.

3.2. Bounds based on the regularity of the loss's gradient

The bounds based on the chaining technique, such as the CMI bound from Asadi et al. (2018) (see Proposition 12 in Section 4), do not fit naturally in the framework presented so far. We are thus motivated to find an alternative setting that naturally gives rise to chained bounds, thus establishing new generalisation results.

As a starting point, let us notice that the main idea behind the CMI bound is to lift the regularity assumption from $x \mapsto \ell(w, x)$ onto $x \mapsto (\ell(w, x) - \ell(w', x))$. A natural guess is that this approach could provide chained bounds also in our general framework, and this is indeed the case (cf. Theorem 22 in Appendix B.1). However, if ℓ is regular enough we can focus on the gradient $\nabla_w \ell(w, x)$ instead. Since this leads to more intuitive and compact statements, we chose to consider this case in the main text.

Assumptions ♣

- The set $\mathcal{W} \subset \mathbb{R}^d$ is convex, compact, and with non-empty interior.
- The function $w \mapsto \ell(w, x)$ is of class C^1 on \mathcal{W} , \mathbb{P}_X -a.s.
- We have $\sup_{(w,x) \in \mathcal{W} \times \mathcal{X}} |\ell(w, x)| < +\infty$ and $\sup_{(w,x) \in \mathcal{W} \times \mathcal{X}} \|\nabla_w \ell(w, x)\| < +\infty$.

Let us stress once more that the above assumptions are not necessary in order to obtain the duality chained-unchained bounds. In Appendix B.1 we discuss a more general setting: \mathcal{W} can be non-convex and with empty interior, ℓ continuous on \mathcal{W} (\mathbb{P}_X -a.s.) and only bounded in expectation.

The chained bounds involve a sequence of finer and finer discretisations of the hypotheses' space, which can be formalised as follows.

4. Note that $w \mapsto \mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W=w})$ is measurable, since both $w \mapsto \mathbb{P}_{S|W=w}$ and $(\mu, \nu) \mapsto \mathfrak{D}(\mu, \nu)$ are Borel measurable (see Appendix F).

Definition 3 (Nets and refining sequences of nets) Given $\varepsilon > 0$, we define an ε -projection on \mathcal{W} as a measurable mapping $\pi : \mathcal{W} \rightarrow \mathcal{W}$ such that $\pi(\mathcal{W})$ has finitely many elements and, for all $w \in \mathcal{W}$, $\|\pi(w) - w\| \leq \varepsilon$. The image $\pi(\mathcal{W})$ is called an ε -net on \mathcal{W} .

Consider a positive, vanishing, decreasing sequence $\{\varepsilon_k\}_{k \in \mathbb{N}}$, and assume that there is a $w_0 \in \mathcal{W}$ such that $\|w - w_0\| \leq \varepsilon_0$ for each $w \in \mathcal{W}$. We call $\{\pi_k(\mathcal{W})\}_{k \in \mathbb{N}}$ an $\{\varepsilon_k\}$ -refining sequence of nets if $\pi_0(\mathcal{W}) = \{w_0\}$ and, for all $k \geq 1$, we have that π_k is a ε_k -projection and $\pi_{k-1} \circ \pi_k = \pi_{k-1}$.

To simplify the notation, for all $w \in \mathcal{W}$ we let $w_k = \pi_k(w)$, and similarly $W_k = \pi_k(W)$ and $\mathcal{W}_k = \pi_k(\mathcal{W})$. Note that for all k , $w_{k'}$ is determined by w_k whenever $k' \leq k$, as $w_{k'} = \pi_{k'}(w_k)$. Moreover, for all $k \geq 1$, $\|w_k - w_{k-1}\| = \|w_k - \pi_{k-1}(w_k)\| \leq \varepsilon_{k-1}$.

The next theorem is the main result of this work. Together with Theorem 2, it establishes the duality between chained and unchained generalisation bounds, which can essentially be obtained by lifting the regularity from the loss onto its gradient.

Theorem 4 Assume \clubsuit and that $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathcal{D}}(\xi)$ wrt \mathbb{P}_S , $\forall w \in \mathcal{W}$. Then, for any $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} ,

$$|\mathcal{G}| = |\mathbb{E}_{\mathbb{P}_W \otimes \mathbb{P}_S}[\mathcal{L}_S(W)] - \mathbb{E}_{\mathbb{P}_W, S}[\mathcal{L}_S(W)]| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W}[\mathcal{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})],$$

where $\mathbb{E}_{\mathbb{P}_W}[\mathcal{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] = \int_{\mathcal{W}} \mathcal{D}(\mathbb{P}_S, \mathbb{P}_{S|W \in \pi_k^{-1}(w)}) d\mathbb{P}_W(w)$.

Proof's sketch Here is a sketch of the proof; see Appendix A.3 for the details. Following the standard chaining argument, we control $\mathcal{L}_s(w)$ by the telescopic sum $\sum_{k \geq 1} (\mathcal{L}_s(w_k) - \mathcal{L}_s(w_{k-1}))$. The upper bound will then follow from the fact that the $\mathcal{R}_{\mathcal{D}}(\xi)$ regularity of $s \mapsto \nabla_w \mathcal{L}_s(w)$ implies the $\mathcal{R}_{\mathcal{D}}(\varepsilon_{k-1}\xi)$ regularity of $w \mapsto (\mathcal{L}_s(w_k) - \mathcal{L}_s(w_{k-1}))$. \blacksquare

Both Theorem 2 and 4 are stated under uniform regularity conditions, in the sense that the value of the regularity's parameter ξ has to be the same for all $w \in \mathcal{W}$. However, we can still achieve generalisation bounds under less strict assumptions. In Appendix B.2 we discuss the case of a measurable map $w \mapsto \xi_w$, such that, for some $p \in [1, +\infty]$, ξ_W is bounded in $L^p(\mathbb{P}_W)$ (or $L^p(\mathbb{P}_{W_k})$, $\forall k \in \mathbb{N}$). Note that choosing $p = +\infty$ brings back the uniform condition.

In a similar spirit, one might try to relax the definition of ε -net, by mimicking the stochastic chaining idea from Zhou et al. (2022). We defer this approach to future work.

4. A few concrete examples: MI and Wasserstein bounds

In the current section we give a few concrete applications of the abstract framework that we have presented so far. We recover some simple generalisation bounds from the literature and establish a novel chained bound, based on the Wasserstein distance.

First, we need to state a few standard definitions.

Definition 5 (Subgaussianity) A real random variable $Z \in L^1$ is ξ -SubGaussian (ξ -SG) if

$$\log \mathbb{E}_{\mathbb{P}_Z}[e^{\lambda Z}] \leq \lambda \mathbb{E}_{\mathbb{P}_Z}[Z] + \frac{\xi^2 \lambda^2}{2}, \quad \forall \lambda > 0.$$

A random vector $V \in \mathbb{R}^q$ is ξ -SG if, for all $v \in \mathbb{R}^q$, $V \cdot v$ is $(\|v\|\xi)$ -SG. Finally, a stochastic process $\{F_w\}_{w \in \mathcal{W}}$ is ξ -SG if, for every pair $(w, w') \in \mathcal{W}^2$, $F_w - F_{w'}$ is a $(\|w - w'\|\xi)$ -SG random variable.

Note that any bounded random variable $Z \in [a, b]$ is $\frac{b-a}{2}$ -SG. Moreover, a normally distributed random variable $Z \sim \mathcal{N}(0, \xi)$ is ξ -SG.

Definition 6 (Lipschitzianity) A function $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ is ξ -Lipschitz on \mathcal{Z} if, for all $z, z' \in \mathcal{Z}$,

$$\|f(z) - f(z')\| \leq \xi d_{\mathcal{Z}}(z, z').$$

Definition 7 (Kullback–Leibler divergence and mutual information) Let μ and ν be two probability measures on \mathcal{Z} . We define the Kullback–Leibler divergence

$$\text{KL}(\nu \parallel \mu) = \begin{cases} \mathbb{E}_{\nu}[\log d\nu/d\mu] & \text{if } \nu \ll \mu; \\ +\infty & \text{otherwise.} \end{cases}$$

For two coupled random variables Z, Z' , the Mutual Information (MI) is defined as

$$I(Z; Z') = \text{KL}(\mathbb{P}_{Z, Z'} \parallel \mathbb{P}_{Z \otimes Z'}).$$

The KL divergence is non-negative, with $\text{KL}(\nu \parallel \mu) = 0$ if, and only if, $\mu = \nu$. Similarly the MI is always non-negative, and null if, and only if, $Z \perp\!\!\!\perp Z'$.

Definition 8 (Wasserstein distance) Given two distributions μ and ν on \mathcal{Z} and fixed $p \geq 1$, their p -Wasserstein distance \mathfrak{W}_p is defined as

$$\mathfrak{W}_p(\mu, \nu) = \inf_{\pi \in \Pi[\mu, \nu]} \mathbb{E}_{(Z, Z') \sim \pi} [d_{\mathcal{Z}}(Z, Z')^p]^{1/p},$$

where $\Pi[\mu, \nu]$ is the set of all probability measures, on $(\mathcal{Z}^2, \Sigma_{\mathcal{Z}} \otimes \Sigma_{\mathcal{Z}})$, with marginals μ and ν .

It can be shown that for $p > p'$ we have $\mathfrak{W}_p(\mu, \nu) \geq \mathfrak{W}_{p'}(\mu, \nu)$, so that in particular \mathfrak{W}_1 is the weakest. For this reason, henceforth we will focus on \mathfrak{W}_1 , which we will simply denote \mathfrak{W} .

Using the concepts that we have just introduced, we can give two simple and concrete examples of \mathfrak{D} -regularity.

Lemma 9 Let $\mathfrak{D}_1 : (\mu, \nu) \mapsto \sqrt{2\text{KL}(\nu \parallel \mu)}$ and $\mathfrak{D}_2 : (\mu, \nu) \mapsto \mathfrak{W}(\mu, \nu)$. Consider a measurable map $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ (with $q \geq 1$). If $f(Z)$ is ξ -SG for $Z \sim \mu \in \mathcal{P}$, then f has regularity $\mathcal{R}_{\mathfrak{D}_1}(\xi)$ wrt μ . If f is ξ -Lipschitz on \mathcal{Z} , then f has regularity $\mathcal{R}_{\mathfrak{D}_2}(\xi)$, wrt any $\mu \in \mathcal{P}$ such that $f \in L^1(\mu)$.

4.1. Standard MI and Wasserstein bounds

We state two simple generalisation bounds that were previously mentioned in the introduction. The proofs that we give leverage the abstract framework of Section 3.1. The first result (Russo and Zou, 2019; Xu and Raginsky, 2017) is an upperbound on \mathcal{G} based on the mutual information between W and S .

Proposition 10 (Standard MI bound) Let $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. If $\ell(w, X)$ is ξ -SG, $\forall w \in \mathcal{W}$, then

$$|\mathcal{G}| \leq \xi \sqrt{\frac{2I(W; S)}{m}}.$$

Proof First, since $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$, $\mathcal{L}_S(w)$ is the average of m independent ξ -SG random variables, so it is (ξ/\sqrt{m}) -SG. In particular, with $\mathfrak{D} : (\mu, \nu) \mapsto \sqrt{2\text{KL}(\nu\|\mu)}$, Lemma 9 shows that $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi/\sqrt{m})$, $\forall w \in \mathcal{W}$. We conclude by Theorem 2 and Jensen's inequality. ■

The next bound is from Lopez and Jog (2018) and is close in spirit to the previous one, as again it tries to measure how much information about S is enclosed in W . However, now the MI is replaced by an expected Wasserstein distance. In order to get an explicit dependence on $1/\sqrt{m}$, we assume that the metric d_S on \mathcal{S} is related to the one on \mathcal{X} via

$$d_S(s, s') = \left(\sum_{i=1}^m d_{\mathcal{X}}(x_i, x'_i)^2 \right)^{1/2}, \quad (1)$$

where $s = \{x_1, \dots, x_m\}$ and $s' = \{x'_1, \dots, x'_m\}$. Note that we do not need $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$.

Proposition 11 (Standard Wasserstein bound) *Suppose that $d_{\mathcal{X}}$ and d_S are related by (1). If, $\forall w \in \mathcal{W}$, $x \mapsto \ell(w, x)$ is ξ -Lipschitz on \mathcal{X} , then*

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \mathbb{E}_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W})].$$

Proof First notice that

$$d_S(s, s') = \left(\sum_{i=1}^m d_{\mathcal{X}}(x_i, x'_i)^2 \right)^{1/2} \geq \frac{1}{\sqrt{m}} \sum_{i=1}^m d_{\mathcal{X}}(x_i, x'_i),$$

where we used the Cauchy-Schwartz inequality. Consequently, $s \mapsto \mathcal{L}_s(w)$ is (ξ/\sqrt{m}) -Lipschitz $\forall w \in \mathcal{W}$. In particular, if we let $\mathfrak{D} : (\mu, \nu) \mapsto \mathfrak{W}(\mu, \nu)$, then $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi/\sqrt{m})$ by Lemma 9. We conclude by Theorem 2. ■

4.2. Chained MI and Wasserstein bounds

As we mentioned in the introduction, one of the main issues with the standard MI bound is that it can easily be vacuous, as it is the case when the learning algorithm defines a deterministic map $\mathcal{S} \rightarrow \mathcal{W}$. To address this issue, Asadi et al. (2018) proposed to build on the chaining technique and established the bound below. The setting here is quite different from the one of the standard MI bound, as the process's subgaussianity takes into account the dependencies between different hypotheses. Letting $\{\varepsilon_k\}_{k \in \mathbb{N}}$ be a vanishing decreasing positive sequence, we consider an $\{\varepsilon_k\}$ -refining sequence of nets $\{\mathcal{W}_k\}_{k \in \mathbb{N}} = \{\pi_k(\mathcal{W})\}_{k \in \mathbb{N}}$ and recall that $W_k = \pi_k(W)$.

Proposition 12 (CMI bound) *Let $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ and \mathcal{W} be a compact set, with an $\{\varepsilon_k\}$ -refining sequence of nets defined on it. Suppose that $w \mapsto \ell(w, x)$ is continuous, for \mathbb{P}_X -almost every x ,⁵ and that $\{\ell(w, X)\}_{w \in \mathcal{W}}$ is a ξ -SG stochastic process. Then we have*

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2I(W_k; S)}.$$

5. Note that in Asadi et al. (2018) the result is stated under a weaker assumption of separability of the process. To avoid introducing further definitions and technicalities in the proofs, we decided to focus on the case of a.s. continuity.

We provide a proof of Proposition 12 within the extended general framework of Appendix B.1, while here we establish a similar result, under the more restrictive assumptions ♣.

Leveraging the machinery developed in Section 3.2, we can expect that lifting the subgaussianity from ℓ to $\nabla_w \ell$ we can find a chained version of the MI bound in Proposition 10. Perhaps unsurprisingly, we simply re-obtain the CMI bound of Proposition 12.

Proposition 13 *Let $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ and assume ♣. If $\nabla_w \ell(w, X)$ is ξ -SG, $\forall w \in \mathcal{W}$, we have that for any $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W}*

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2I(W_k; S)}.$$

Proof As in the proof of Proposition 10, we have that $\nabla_w \mathcal{L}_S(w)$ is (ξ/\sqrt{m}) -SG, $\forall w \in \mathcal{W}$. In particular, by Lemma 9 we have that $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi/\sqrt{m})$, $\forall w \in \mathcal{W}$, where $\mathfrak{D} : (\mu, \nu) \mapsto \sqrt{2\text{KL}(\nu\|\mu)}$. Hence, we conclude by Theorem 4 and Jensen's inequality. ■

The next lemma shows that, under the assumptions ♣, Propositions 12 and 13 are equivalent.

Lemma 14 *Under the assumptions ♣, the stochastic process $(\ell(w, X))_{w \in \mathcal{W}}$ is ξ -SG if, and only if, $\nabla_w \ell(w, X)$ is a ξ -SG vector for all $w \in \mathcal{W}$.*

Once again, the main point of the abstract framework presented so far is to underline a duality: to each bound based on the \mathfrak{D} -regularity of the loss corresponds a chained bound based on the \mathfrak{D} -regularity of its gradient. We can hence apply this idea to the standard Wasserstein bound of Proposition 11 and obtain its chained counterpart, which is a novel result.

Proposition 15 (Chained Wasserstein bound) *Let $d_{\mathcal{X}}$ and d_S be related by (1). Under the assumptions ♣, suppose that $x \mapsto \nabla_w \ell(w, x)$ is ξ -Lipschitz on \mathcal{X} , $\forall w \in \mathcal{W}$. Then, for any $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} ,*

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{\mathcal{W}}} [\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W_k})].$$

Proof Let $\mathfrak{D} : (\mu, \nu) \mapsto \mathfrak{W}(\mu, \nu)$. Proceeding as in the proof of Proposition 11, we have that $\nabla_w \mathcal{L}_S(w)$ is (ξ/\sqrt{m}) -Lipschitz, $\forall w \in \mathcal{W}$. In particular, by Lemma 9, $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi/\sqrt{m})$, $\forall w \in \mathcal{W}$. Hence, we conclude by Theorem 4. ■

We conclude by recalling once more that, in our framework, any bound based on the regularity of ℓ gives rise to a chained bound. We refer to Table 1 in the appendix for several explicit examples.

5. A chained PAC-Bayesian bound

The framework introduced in Section 3 focuses on the *backward-channel* information-theoretic setting. However, the chaining ideas behind Theorem 4 can fit in a broader context. As an example, we discuss here a PAC-Bayesian result. Although Audibert and Bousquet (2004) have already combined the PAC-Bayesian approach with the chaining technique, their use of an auxiliary sample and of the average distance between nets makes their bounds conceptually different from ours.

The PAC-Bayesian bounds are algorithmic-dependent upper bounds on the expected generalisation error $\mathbb{E}_{\mathbb{P}_{W|S}}[g_S(W)]$ of stochastic classifiers (McAllester, 1998), holding with high probability on the random draw of the training sample S (see Guedj (2019) and Alquier (2021) for recent introductory overviews). They share the same underlying idea with the information-theoretic bounds: the less $\mathbb{P}_{W|S}$ is dependent on S , the better the algorithm generalises. However, in the PAC-Bayesian setting we compare $\mathbb{P}_{W|S}$ not with the marginal \mathbb{P}_W , but rather with a fixed probability measure \mathbb{P}_W^* , which can be chosen arbitrarily but without making use of the training sample S .

We state here a very simple classical PAC-Bayesian result from Catoni (2009).

Proposition 16 *Assume that ℓ is bounded in $[-\xi, \xi]$. Let \mathbb{P}_W^* be a fixed probability measure on \mathcal{W} , chosen independently of S . Fix $\delta \in (0, 1)$ and $\lambda > 0$. Then, with probability $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ larger than $1 - \delta$ on the draw of S , we have*

$$\mathbb{E}_{\mathbb{P}_{W|S}}[g_S(W)] \leq \frac{\xi}{\sqrt{2m}} \left(\lambda + \frac{\text{KL}(\mathbb{P}_{W|S} \|\mathbb{P}_W^*) + \log \frac{1}{\delta}}{\lambda} \right).$$

A chained version of the above can be obtained by lifting the boundedness hypothesis from ℓ to $\nabla_w \ell$. This is quite peculiar, as most PAC-Bayesian bounds hold for bounded loss functions $\ell \subseteq [-\xi, \xi]$.

Proposition 17 *Under the assumptions ♣, consider a $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} and assume that $\nabla_w \ell$ is bounded in $[-\xi, \xi]$. Let \mathbb{P}_W^* be a fixed probability measure on \mathcal{W} , chosen independently of S . Fix two sequences $\{\delta_k\}_{k \in \mathbb{N}}$ and $\{\lambda_k\}_{k \in \mathbb{N}}$, such that $\delta_k \in (0, 1)$ and $\lambda_k > 0$ for all k . Assume that $\sum_{k \in \mathbb{N}} \delta_k = \delta \in (0, 1)$. Then, with probability $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ larger than $1 - \delta$ on the draw of S , we have*

$$\mathbb{E}_{\mathbb{P}_{W|S}}[g_S(W)] \leq \frac{\xi}{\sqrt{2m}} \left(2 \sqrt{\log \frac{1}{\delta_0}} + \sum_{k=1}^{\infty} \varepsilon_{k-1} \left(\lambda_k + \frac{\text{KL}(\mathbb{P}_{W_k|S} \|\mathbb{P}_{W_k}^*) + \log \frac{1}{\delta_k}}{\lambda_k} \right) \right).$$

The PAC-Bayesian bound in Proposition 16 is infinite for a deterministic algorithm (that is when $\mathbb{P}_{W|S=s}$ is a Dirac delta for all $s \in S$). Remarkably, for suitable coefficients λ_k , δ_k , and ε_k , the chained bound of Proposition 17 is always finite, since all the terms $\text{KL}(\mathbb{P}_{W_k|S} \|\mathbb{P}_{W_k}^*)$ are bounded by $\log |\mathcal{W}_k|$. However, the best choice of the parameters λ and λ_k is delicate, as it cannot depend on S (and hence on the KL term). We refer to Appendix C for further discussion on this last point.

6. Comparison of chained and unchained bounds

Having established the duality, we are left with the Hamletic question: *chained or unchained, what is the best?* First, we notice that the requirements for the chained bounds are somewhat stronger.

Lemma 18 *Under the assumptions ♣, let ε_0 and w_0 be such that $\|w - w_0\| \leq \varepsilon_0, \forall w \in \mathcal{W}$. Assume that $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt $\mathbb{P}_S, \forall w \in \mathcal{W}$, and define $\mathcal{L}_s(w) = \mathcal{L}_s(w) - \mathcal{L}_s(w_0)$ and $\hat{\mathcal{G}} = \mathbb{E}_{W \otimes S}[\mathcal{L}_S(W)] - \mathbb{E}_{W,S}[\mathcal{L}_S(W)]$. Then, $\hat{\mathcal{G}} = \mathcal{G}$, and $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\varepsilon_0 \xi)$, wrt \mathbb{P}_S and $\forall w \in \mathcal{W}$.*

Hence, whenever we derive a chained bound $|\mathcal{G}| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})]$ in our framework, we can always state an unchained counterpart in the form $|\mathcal{G}| \leq \varepsilon_0 \xi \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W})]$. Nevertheless, the next result shows that conditioning on W_k instead of W can often be helpful.

Lemma 19 *Assume that $\mu \mapsto \mathfrak{D}(\mathbb{P}_S, \mu)$ is convex. For any $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} , the sequence $\{\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})]\}_{k \in \mathbb{N}}$ is non-decreasing and, $\forall k \in \mathbb{N}$, we have*

$$\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] \leq \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W})].$$

$\text{KL}(\nu \parallel \mu)$ is convex in both ν and μ (Erven and Harremoës, 2014), and the same holds for $\mathfrak{W}(\mu, \nu)$ (Villani, 2009). Thus, $I(W_k; S) \leq I(W; S)$ ⁶ and $\mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] \leq \mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W})]$.

Lemma 19 alone is not enough to ensure that the chained bound is tighter than its unchained counterpart. However, if \mathbb{P}_W is very concentrated on a tiny region of \mathcal{W} , so that S is almost independent of W_k up to a small scale (*i.e.*, large k), then one can expect the chained result to be the tightest. We will clarify this intuition by means of two simple toy examples. Since Asadi et al. (2018) have already shown that the CMI bound can be much tighter than the MI one, here the focus is on the Wasserstein bounds.

6.1. Comparison of the chained and unchained Wasserstein bounds

In the following we denote by \mathcal{B}_ℓ the standard Wasserstein bound (Proposition 11) and by $\mathcal{B}_{\nabla\ell}$ its chained counterpart (Proposition 15). For simplicity, we mainly focus on the case $m = 1$, so that we can write $s = x$ and $\mathcal{G} = \mathbb{E}_{\mathbb{P}_{W \otimes X}}[\ell(W, X)] - \mathbb{E}_{\mathbb{P}_{W, X}}[\ell(W, X)]$.

Example 1 Let $\mathcal{W} = \mathcal{X} = [-1, 1]$, $\ell(w, x) = \frac{1}{2}(w - x)^2$, and $\varepsilon_k = 2^{-k}$, for $k \in \mathbb{N}$. We can find mappings π_k that define an $\{\varepsilon_k\}$ -refining sequence of nets, with $\mathcal{W}_k = \{2^{1-k}j : j \in [-2^{k-1} : 2^k - 1]\}$, where $[a : b] = [a, b] \cap \mathbb{Z}$. Fix $k^* \in \mathbb{N}$ and define $\theta = 2^{-k^*}$. Let X be uniformly distributed on $(-\theta, \theta)$. We choose an algorithm that, given x , selects the w minimising $\ell(w, x)$. This means that $\mathbb{P}_{W|X=x} = \delta_x$, where δ_x is the Dirac measure centred on x . Note that $\nabla_w \ell$ is 1-Lipschitz and ℓ is 2-Lipschitz (on \mathcal{X} , uniformly on \mathcal{W}). However, thanks to Lemma 18 we know that we can consider the loss $\tilde{\ell}(w, x) = \ell(w, x) - \frac{x^2}{2}$, which leads to the same generalisation and is 1-Lipschitz.

In this simple example, we can compute exactly everything we need (see Appendix E.1):

$$|\mathcal{G}| = \frac{1}{3}\theta^2 \simeq 0.33\theta^2; \quad \frac{1}{2}\mathcal{B}_\ell = \mathcal{B}_{\tilde{\ell}} = \frac{2}{3}\theta \simeq 0.67\theta; \quad \mathcal{B}_{\nabla\ell} = \frac{247}{105}\theta^2 \simeq 2.35\theta^2.$$

Note that, as θ decreases, \mathbb{P}_W becomes more and more concentrated, since W lies with probability 1 in $(-\theta, \theta)$. In particular, X and W_k are independent for $k \leq k^* = -\log_2 \theta$, and so the first k^* terms in the chaining sum are null. For this reason, $\mathcal{B}_{\nabla\ell}$ captures the right behaviour $O(\theta^2)$ of \mathcal{G} for $\theta \rightarrow 0$, which is not the case for \mathcal{B}_ℓ and $\mathcal{B}_{\tilde{\ell}}$.

Quite interestingly, it is possible to explicitly evaluate the CMI bound (\mathcal{B}_{CMI}) as well. We find $\mathcal{B}_{\text{CMI}} \simeq 3.50\theta$, meaning that in this example the chained MI bound fails to capture the right behaviour of \mathcal{G} as $\theta \rightarrow 0$. We refer to Section 7 for a few comments about this. On the other hand, the unchained MI bound is infinite, since W is a deterministic function of X .

Finally, if we consider a larger random sample $S = \{X_1, \dots, X_m\}$, with $m > 1$, we still have that the ratio $\mathcal{B}_{\nabla\mathcal{L}}/\mathcal{B}_{\mathcal{L}}$ (between the chained and unchained Wasserstein bounds) vanishes as $O(\theta)$ for $\theta \rightarrow 0$. Again, this is a consequence of the fact that S and W_k are independent for $k \leq k^*$, since W is the empirical average $\sum_i X_i/m$ and lies in $(-\theta, \theta)$ with probability 1.

6. This can also be seen as a trivial consequence of the data-processing inequality.

Example 2 This toy model is inspired by Example 1 in [Asadi et al. \(2018\)](#). Let $\mathcal{W} = \{w \in \mathbb{R}^2 : \|w\| = 1\}$ ⁷ and $\mathcal{X} = \mathbb{R}^2$. Fix $a > 0$ and let $X \sim \mathcal{N}((a, 0), \text{Id})$, a normal distribution centered in $(a, 0)$, with the identity matrix as covariance. The algorithm aims at finding the direction of the mean of X (that is $(1, 0)$), by choosing the w that minimises the loss $\ell(w, x) = -w \cdot x$. Let $w_0 = (1, 0)$ and $\varepsilon_0 = 4$. For $k \geq 1$, let $\mathcal{W}_k = \{w = (\cos \frac{2\pi j}{2^k}, \sin \frac{2\pi j}{2^k}) : j \in [-2^{k-1} : 2^{k-1} - 1]\}$ and $\varepsilon_k = 4/2^k$. We can then easily find projections π_k that make $\{\mathcal{W}_k\}_{k \in \mathbb{N}}$ an $\{\varepsilon_k\}$ -sequence of refining nets. Both ℓ and $\nabla_w \ell$ are 1-Lipschitz in \mathcal{X} , $\forall w \in \mathcal{W}$. Although it is hard to find the analytic expressions for \mathcal{B}_ℓ and $\mathcal{B}_{\nabla \ell}$, we can study their asymptotic behaviour for $a \rightarrow \infty$. In this limit, \mathbb{P}_W becomes highly concentrated around $(0, 1)$, as it tends towards a Dirac delta. So, for a large enough we expect the chained bound to be the tightest. Indeed, we find

$$|\mathcal{G}| = \Theta(1/a); \quad \mathcal{B}_\ell = \Theta(1); \quad \mathcal{B}_{\nabla \ell} = O((\log a - \log \log a)/a).^8$$

Up to logarithmic factors, $\mathcal{B}_{\nabla \ell}$ can capture the correct behaviour of $|\mathcal{G}|$ as $a \rightarrow \infty$.

As a final remark, note that in this example the loss ℓ is not Lipschitz on \mathcal{W} , uniformly on \mathcal{X} , and so the *forward-channel* Wasserstein bound from [Wang et al. \(2019\)](#) does not apply.⁹

6.2. High concentration is not always enough

In both the previous examples, the chained bound was much tighter than its unchained counterpart when \mathbb{P}_W was highly concentrated in a small neighbourhood U of a single point w_* . In particular, if 2ε is the diameter of U , we can expect that just knowing that $W \in U$ is not informative up to a length-scale of order ε . However, this can easily fail when W concentrates around two far apart points (say w_1 and w_2). Indeed, if for small k we already have that $\pi_k(w_1) \neq \pi_k(w_2)$, knowing that the chosen hypothesis is next to w_1 might bring a lot of information about S . On the other hand, one can still imagine situations in which there are multiple points around which W concentrates, yet which one is the nearest to the chosen hypothesis is not informative about the sample.

In [Appendix E.1.1](#), we discuss a high-dimensional version of [Example 1](#), where W does not concentrate around a single point, but in a thin neighbourhood of a one-dimensional line. We show that when θ (the parameter describing the size of the support of W) has the right scaling with the dimension d of \mathcal{W} , the ratio $\mathcal{B}_{\nabla \ell}/\mathcal{B}_\ell$ vanishes as $d \rightarrow \infty$.

7. Comparison between MI and Wasserstein bounds

We conclude this paper with a few comments on the relation between the MI-based ([Propositions 10 and 13](#)) and the Wasserstein-based bounds ([Propositions 11 and 15](#)). The problem comes down to comparing the KL divergence with the 1-Wasserstein distance, a task closely related to transportation-cost inequalities (see [Raginsky and Sason \(2013\)](#) for a pedagogical overview). Let μ be a probability measure on the Polish space $(\mathcal{Z}, \Sigma_{\mathcal{Z}})$. For $\eta > 0$, μ is said to satisfy a L^1 transport-cost inequality with constant η (in short $\mu \in T_1(\eta)$) if, for any $\nu \ll \mu$, $\mathfrak{W}(\mu, \nu) \leq \sqrt{2\eta \text{KL}(\nu \|\mu)}$. Hence, whenever $\mathbb{P}_S \in T_1(1)$ we are assured that each one of the two Wasserstein-based bounds is

7. This is not a convex set. However, one can either suitably extend ℓ to the unit disk in \mathbb{R}^2 , or easily check that the hypotheses of the extended framework of [Theorem 22](#) in [Appendix B.1](#) are verified (see [Section E.2](#)).

8. Here $f = O(g)$ stands for $\lim_{a \rightarrow \infty} |f(a)/g(a)| < \infty$, while by $f = \Theta(g)$ we mean that $f = O(g)$ and $g = O(f)$.

9. Of course, $\ell(w, x) = -w \cdot x/\|x\|$ would bring the same algorithm and is 1-Lipschitz in w . However this is just due to the radial symmetry. Changing the problem slightly and considering for instance $\ell(w, x) = -w \cdot \psi(x)$, for a general 1-Lipschitz map $\psi : \mathcal{X} \rightarrow \mathcal{X}$, would not allow to easily find an equivalent loss that is 1-Lipschitz in w .

tighter than the corresponding MI-based one. For instance, this is the case when \mathbb{P}_S is a multivariate normal whose covariance matrix is the identity (Talagrand, 1996), as in Example 2. However, there is a price to pay: whenever the L^1 transport-inequality holds, then Lipschitzianity is stronger than subgaussianity. More precisely, Bobkov and Götze (1999) showed that $\mu \in T_1(1)$ if, and only if, for every ξ -Lipschitz function $f : \mathcal{Z} \rightarrow \mathbb{R}$, $f(Z)$ is ξ -subgaussian for $Z \sim \mu$.

It is worth noticing that, if the size of the support of X is particularly small, the Wasserstein bounds can be much tighter than the MI ones. This is for instance the case in Example 1, where the length-scale of the support of X is given by θ . There, the chained Wasserstein bound goes as θ^2 . A factor θ is brought by the chaining technique, which allows us to neglect the contributions of the larger length-scales, whilst the other factor θ is due to the use of the Wasserstein distance, which intrinsically takes into account the considered length-scale. In contrast, since the MI is scale-invariant, the CMI bound has only a linear dependence in θ coming from the chaining method.

7.1. Scaling with the sample size

It is worth mentioning the different roles that the factor $1/\sqrt{m}$ plays in the MI and the Wasserstein bounds. In the MI bound this scaling is linked to concentration properties, since it comes from the fact that the average of m independent ξ -SG random variables is (ξ/\sqrt{m}) -SG. The requirement that S is made of independent draws is hence essential in this case. On the other hand, in the Wasserstein bound the factor $1/\sqrt{m}$ has a merely geometric origin and follows from the relation (1) between the metrics $d_{\mathcal{X}}$ and d_S . In particular, an alternative choice of d_S might yield a different factor in front of the bound, but also change the scaling with m of the Wasserstein distance. A priori, it is not easy to say which d_S would bring the tightest bound. Once more, let us stress that the Wasserstein bound does not require that $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. Indeed, $\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W})$ will take into account the dependencies between the training inputs, and we can expect it to scale poorly with m if the different X_i in S are strongly correlated. However, even in the case of independent X_i , it is hard to say in general what is the exact dependence with m , for both $I(W; S)$ and $\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W})$.

As a final remark about the case $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$, just by looking at \mathbb{P}_X it is sometimes possible to establish that both the standard and chained Wasserstein bounds are tighter than their MI counterparts, no matter the size of the training dataset and the choice of the algorithm. To this purpose, we can again exploit some classical results on the transport-cost inequalities (Raginsky and Sason, 2013; Gozlan and Léonard, 2010). For a probability measure μ , we say that $\mu \in T_2(1)$ if, for any $\nu \ll \mu$, $\mathfrak{W}_2(\mu, \nu) \leq \sqrt{2\text{KL}(\nu||\mu)}$. It is known that $\mu \in T_2(1)$ implies that $\mu^{\otimes m} \in T_2(1)$, $\forall m \geq 1$. In particular, if $\mathbb{P}_X \in T_2(1)$, then we are ensured that $\mathbb{P}_S = \mathbb{P}_X^{\otimes m} \in T_2(1)$. Since $\mathfrak{W} = \mathfrak{W}_1 \leq \mathfrak{W}_2$, $\mathbb{P}_X \in T_2(1)$ actually implies $\mathbb{P}_S \in T_1(1)$, which (as we discussed the beginning of this section) means that each Wasserstein-based bound is tighter than the corresponding MI-based one.

8. Conclusion

We introduced a general framework allowing us to derive new generalisation results leveraging on the chaining technique. By doing so, under suitable regularity conditions we established a duality between chained and unchained generalisation bounds. Although the chained bounds usually come at the price of stricter assumptions, sometimes they better capture the loss function’s behaviour, especially in cases where the hypothesis distribution is highly concentrated. We hence believe that combining the chaining method with other information-theoretic techniques is a promising direction in order to tighten the bounds on the generalisation error.

In this work we have mainly focused on the *backward-channel* information-theoretic perspective, as we believe that it combines naturally with the chaining on the hypotheses' space. However, the chained PAC-Bayesian result that we presented is an example of a *forward-channel* bound, as it considers the distribution of the hypotheses, conditioned on the sample. A future direction of study could be to extend our general framework to include *forward-channel* bounds. We believe this should not present major technical difficulties and might bring new interesting results.

Although information-theoretic bounds are usually hard to evaluate in practice, recent works have derived computable analytic bounds for specific algorithms, such as Langevin dynamics or stochastic gradient descent, by upper-bounding information-theoretic generalisation results. We believe that combining these ideas with the chaining technique is a venue worth exploring.

Acknowledgments

Eugenio Clerico is partly supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the grant EP/R513295/1 (DTP scheme). Arnaud Doucet is partly supported by the EPSRC grant EP/R034710/1. He also acknowledges the support of the UK Defence Science and Technology Laboratory (DSTL) and EPSRC under grant EP/R013616/1. This is part of the collaboration between US DOD, UK MOD, and UK EPSRC, under the Multidisciplinary University Research Initiative.

References

- P. Alquier. User-friendly introduction to PAC-Bayes bounds. *arXiv:2110.11216*, 2021.
- G. Aminian, L. Toni, and M. R. D. Rodrigues. Information-theoretic bounds on the moments of the generalization error of learning algorithms. *arXiv:2102.02016*, 2021.
- M. Anthony and P. L. Bartlett. *Neural Network Learning - Theoretical Foundations*. Cambridge University Press, 2002.
- A. R. Asadi and E. Abbe. Chaining meets chain rule: Multilevel entropic regularization and training of neural nets. *JMLR*, 21, 2020.
- A. R. Asadi, E. Abbe, and S. Verdú. Chaining mutual information and tightening generalization bounds. *NeurIPS*, 2018.
- J.-Y. Audibert and O. Bousquet. PAC-Bayesian generic chaining. *NeurIPS*, 2004.
- M. Belkin, S. Ma, and S. Mandal. To understand deep learning we need to understand kernel learning. *ICML*, 2018.
- S. G. Bobkov and F. Götze. Exponential integrability and transportation cost related to logarithmic sobolev inequalities. *Journal of Functional Analysis*, 163(1), 1999.
- O. Bousquet and A. Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2, 2002.
- O. Bousquet, S. Boucheron, and G. Lugosi. *Introduction to Statistical Learning Theory*. Springer, 2004.
- Y. Bu, S. Zou, and V. V. Veeravalli. Tightening mutual information based bounds on generalization error. *ISIT*, 2019.
- O. Catoni. A PAC-Bayesian approach to adaptive classification. *Preprint LPMA*, 840, 2009.
- E. A. Cooper and H. Farid. A toolbox for the radial and angular marginalization of bivariate normal distributions. *arXiv:2005.09696*, 2020.
- M. D. Donsker and S. R. S. Varadhan. Asymptotic evaluation of certain Markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2), 1983.
- R. M. Dudley. The sizes of compact subsets of Hilbert space and continuity of Gaussian processes. *Journal of Functional Analysis*, 1(3), 1967.
- R. M. Dudley. *Real Analysis and Probability*. Cambridge University Press, 2002.
- C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9, 2014.
- T. Erven and P. Harremoës. Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, 60(7), 2014.

- A. R. Esposito, M. Gastpar, and I. Issa. Generalization error bounds via Rényi-, f -divergences and maximal leakage. *IEEE Transactions on Information Theory*, 67(3), 2021.
- N. Gozlan and C. Léonard. Transport inequalities. A survey. *Markov Processes and Related Fields*, 16, 2010.
- B. Guedj. A primer on PAC-Bayesian learning. *Proceedings of the Second Congress of the French Mathematical Society*, 2019.
- A. Guntuboyina, S. Saha, and G. Schiebinger. Sharp inequalities for f -divergences. *IEEE Transactions on Information Theory*, 60(1), 2014.
- M. Haghifam, J. Negrea, A. Khisti, D. M. Roy, and G. K. Dziugaite. Sharpened generalization bounds based on conditional mutual information and an application to noisy, iterative algorithms. *NeurIPS*, 2020.
- F. Hellström and G. Durisi. Generalization bounds via information density and conditional information density. *IEEE Journal on Selected Areas in Information Theory*, 1(3), 2020.
- W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301), 1963.
- A. S. Kechris. *Classical Descriptive Set Theory*. Springer-Verlag, 1995.
- E. L. Lehmann and G. Casella. *Theory of Point Estimation*. Springer-Verlag, New York, NY, USA, second edition, 1998.
- M. Liero, A. Mielke, and G. Savaré. Optimal entropy-transport problems and a new Hellinger–Kantorovich distance between positive measures. *Inventiones Mathematicae*, 211, 2018.
- A. T. Lopez and V. Jog. Generalization error bounds using Wasserstein distances. *IEEE Information Theory Workshop (ITW)*, 2018.
- G. Marsaglia, B. Narasimhan, and A. Zaman. The distance between random points in rectangles. *Communications in Statistics - Theory and Methods*, 19, 1990.
- D. A. McAllester. Some PAC-Bayesian theorems. *COLT*, 1998.
- D. A. McAllester. PAC-Bayesian model averaging. *COLT*, 1999.
- J. Negrea, M. Haghifam, G. K. Dziugaite, A. Khisti, and D. M. Roy. Information-theoretic generalization bounds for SGLD via data-dependent estimates. *NeurIPS*, 2019.
- G. Neu, G. K. Dziugaite, M. Haghifam, and D. M. Roy. Information-theoretic generalization bounds for stochastic gradient descent. *COLT*, 2021.
- D. P. Palomar and S. Verdú. Lautum information. *IEEE Transactions on Information Theory*, 54(3), 2008.

- M. Raginsky and I. Sason. Concentration of measure inequalities in information theory, communications, and coding. *Foundations and Trends in Communications and Information Theory*, 10 (1-2), 2013.
- B. Rodríguez-Gálvez, G. Bassi, R. Thobaben, and M. Skoglund. On random subset generalization error bounds and the stochastic gradient Langevin dynamics algorithm. *IEEE Information Theory Workshop (ITW)*, 2020.
- B. Rodríguez-Gálvez, G. Bassi, R. Thobaben, and M. Skoglund. Tighter expected generalization error bounds via Wasserstein distance. *arXiv:2101.09315*, 2021.
- D. Russo and J. Zou. How much does your data exploration overfit? controlling bias via information usage. *IEEE Transactions on Information Theory*, 66(1), 2019.
- S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.
- T. Steinke and L. Zakyntinou. Reasoning about generalization via conditional mutual information. *COLT*, 2020.
- M. Talagrand. Transportation cost for Gaussian and other product measures. *Geometric and Functional Analysis*, 6(3), 1996.
- M. Talagrand. *The Generic Chaining: Upper and Lower Bounds of Stochastic Processes*. Springer-Verlag, 2005.
- M. Talagrand. *Upper and Lower Bounds for Stochastic Processes: Modern Methods and Classical Problems*. Springer-Verlag, 2014.
- R. van Handel. *Probability in High Dimensions*, 2016. [Online; accessed on 02/2022.] <https://web.math.princeton.edu/~rvan/APC550.pdf>.
- V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.
- R. Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.
- C. Villani. *Optimal Transport – Old and New*. Springer, 2009.
- H. Wang, M. Diaz, J. C. S. Santos Filho, and F. P. Calmon. An information-theoretic view of generalization via Wasserstein distance. *ISIT*, 2019.
- A. D. Wyner. A definition of conditional mutual information for arbitrary ensembles. *Information and Control*, 38(1), 1978.
- A. Xu and M. Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. *NeurIPS*, 2017.
- C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3), 2021.
- R. Zhou, C. Tian, and T. Liu. Stochastic chaining and strengthened information-theoretic generalization bounds. *arXiv:2201.12192*, 2022.

Appendix A. Omitted proofs of Sections 3 and 4

Here $(\mathcal{Z}, d_{\mathcal{Z}})$ is a separable complete metric space, with Borel σ -algebra $\Sigma_{\mathcal{Z}}$ induced by the metric. $\mathcal{W} \times \mathcal{Z}$ is endowed with the product σ -algebra $\Sigma_{\mathcal{W}} \otimes \Sigma_{\mathcal{Z}}$. \mathcal{P} denotes the space of probability measures on \mathcal{Z} and is endowed with the σ -algebra induced by the topology of weak convergence.

A.1. Proof of Lemma 9

Lemma 9 *Let $\mathfrak{D}_1 : (\mu, \nu) \mapsto \sqrt{2\text{KL}(\nu\|\mu)}$ and $\mathfrak{D}_2 : (\mu, \nu) \mapsto \mathfrak{W}(\mu, \nu)$. Consider a measurable map $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ (with $q \geq 1$). If $f(Z)$ is ξ -SG for $Z \sim \mu \in \mathcal{P}$, then f has regularity $\mathcal{R}_{\mathfrak{D}_1}(\xi)$ wrt μ . If f is ξ -Lipschitz on \mathcal{Z} , then f has regularity $\mathcal{R}_{\mathfrak{D}_2}(\xi)$, wrt any $\mu \in \mathcal{P}$ such that $f \in L^1(\mu)$.*

Proof First, notice that Lemmas 28 and 29 ensure that both \mathfrak{D}_1 and \mathfrak{D}_2 are measurable, as required by Definition 1.

Assume that $f(Z)$ is ξ -SG for $Z \sim \mu$. Then, by definition $f \in L^1(\mu)$. Fix ν such that $f \in L^1(\nu)$ and $\text{Supp}(\nu) \subseteq \text{Supp}(\mu)$. If $q = 1$, the Donsker-Varadhan representation of KL (Donsker and Varadhan, 1983) and subgaussianity yield

$$\text{KL}(\nu\|\mu) \geq \sup_{\lambda \in \mathbb{R}} \lambda(\mathbb{E}_{\nu}[f(Z)] - \mathbb{E}_{\mu}[f(Z)]) - \lambda^2 \xi^2 / 2 = \frac{(\mathbb{E}_{\mu}[f(Z)] - \mathbb{E}_{\nu}[f(Z)])^2}{2\xi^2},$$

from which the \mathfrak{D}_1 -regularity of f follows immediately. The case of a generic $q > 1$ is trivial, since $v \cdot f(Z)$ is $(\xi\|v\|)$ -SG by Definition 5, for all $v \in \mathbb{R}^q$.

Now, let $f \in L^1(\mu)$ be ξ -Lipschitz. If $q = 1$, let π be any coupling with marginals μ and ν . We have that

$$|\mathbb{E}_{\mu}[f(Z)] - \mathbb{E}_{\nu}[f(Z)]| = |\mathbb{E}_{(Z, Z') \sim \pi}[f(Z) - f(Z')]| \leq \xi \mathbb{E}_{(Z, Z') \sim \pi}[d(Z, Z')].$$

The \mathfrak{D}_2 -regularity can be established by taking the inf among all the couplings π with marginals μ and ν . The case $q > 1$ follows from the fact that $z \mapsto v \cdot f(z)$ is $(\xi\|v\|)$ -Lipschitz. ■

A.2. Proof of Theorem 2

Theorem 2 is equivalent to the following result.

Theorem 20 *Consider a measurable map $F : \mathcal{W} \times \mathcal{Z} \rightarrow \mathbb{R}$, such that $z \mapsto F(w, z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt \mathbb{P}_Z and for all $w \in \mathcal{W}$. Then we have*

$$|\mathbb{E}_{\mathbb{P}_{\mathcal{W} \otimes \mathcal{Z}}}[F(W, Z)] - \mathbb{E}_{\mathbb{P}_{\mathcal{W}, Z}}[F(W, Z)]| \leq \xi \mathbb{E}_{\mathbb{P}_{\mathcal{W}}}[\mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W})].$$

Proof First, note that $\text{Supp}(\mathbb{P}_{Z|W=w}) \subseteq \text{Supp}(\mathbb{P}_Z)$ by Lemma 30 and $\mathbb{E}_{\mathbb{P}_{Z|W=w}}[|F(w, Z)|] < \infty$, \mathbb{P}_W -a.s., since $\mathbb{E}_{\mathbb{P}_{\mathcal{W}, Z}}[|F(W, Z)|] < +\infty$. In particular, for \mathbb{P}_W -almost every $w \in \mathcal{W}$ we have that

$$|\mathbb{E}_{\mathbb{P}_Z}[F(w, Z)] - \mathbb{E}_{\mathbb{P}_{Z|W=w}}[F(w, Z)]| \leq \xi \mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W=w}).$$

Then the conclusion follows by taking the expectation wrt \mathbb{P}_W and using Jensen's inequality. ■

A.3. Proof of Theorem 4

Theorem 4 follows from the next result, which is a direct corollary of Theorem 22 and Lemma 23, proved in Appendix B.1.

Theorem 21 *Let \mathcal{W} be a compact convex subset of \mathbb{R}^d with non-empty interior. Consider a measurable map $F : \mathcal{W} \times \mathcal{Z} \rightarrow \mathbb{R}$, such that $w \mapsto F(w, z)$ is C^1 , \mathbb{P}_Z -a.s. Assume that $\sup_{(w,z) \in \mathcal{W} \times \mathcal{X}} |F(w, z)| < +\infty$ and $\sup_{(w,z) \in \mathcal{W} \times \mathcal{X}} |\nabla_w F(w, z)| < +\infty$. If $z \mapsto \nabla_w F(w, z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt \mathbb{P}_Z , $\forall w \in \mathcal{W}$, then we have that for any $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W}*

$$|\mathbb{E}_{\mathbb{P}_{\mathcal{W} \otimes \mathcal{Z}}} [F(W, Z)] - \mathbb{E}_{\mathbb{P}_{\mathcal{W}, Z}} [F(W, Z)]| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{\mathcal{W}}} [\mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W_k})].$$

Proof By Lemma 23, the regularity of $\nabla_w F$ implies that the map $z \mapsto (F(w, z) - F(w', z))$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi \|w - w'\|)$, wrt \mathbb{P}_Z and for all $w, w' \in \mathcal{W}$. We conclude by Theorem 22. \blacksquare

A.4. Proof of Lemma 14

Lemma 14 *Under the assumptions \clubsuit , the stochastic process $(\ell(w, X))_{w \in \mathcal{W}}$ is ξ -SG if, and only if, $\nabla_w \ell(w, X)$ is a ξ -SG vector for all $w \in \mathcal{W}$.*

Proof First, notice that, without loss of generality, we can consider the case of a one-dimensional $\mathcal{W} \subseteq \mathbb{R}$. Indeed, if \mathcal{W} is higher dimensional, for any two given points w and w' , we can always restrict to a line connecting them, making the problem 1D. Moreover, letting $\bar{\ell}(w, x) = \ell(w, x) - \mathbb{E}_{\mathbb{P}_X}[\ell(w, X)]$ we have that the assumptions in \clubsuit ensure that $\nabla_w \bar{\ell} = \nabla_w \ell - \mathbb{E}_{\mathbb{P}_X}[\nabla_w \ell]$. So, we just need to show that the lemma holds for $\bar{\ell}$.

Now, let $\bar{\ell}$ be a ξ -SG process, so that for $\varepsilon \neq 0$ and $\lambda \in \mathbb{R}$

$$\mathbb{E}_{\mathbb{P}_X} [e^{\lambda(\bar{\ell}(w+\varepsilon, X) - \bar{\ell}(w, X))/\varepsilon}] \leq e^{\frac{\lambda^2}{2\varepsilon^2} \xi^2 \varepsilon^2} = e^{\frac{\lambda^2 \xi^2}{2}}.$$

In particular, by Fatou's lemma we have

$$\mathbb{E}_{\mathbb{P}_X} [e^{\lambda \partial_w \bar{\ell}(w, X)}] = \mathbb{E}_{\mathbb{P}_X} \left[\lim_{\varepsilon \rightarrow 0} e^{\lambda \frac{\bar{\ell}(w+\varepsilon, X) - \bar{\ell}(w, X)}{\varepsilon}} \right] \leq \liminf_{\varepsilon \rightarrow 0} \mathbb{E}_{\mathbb{P}_X} \left[e^{\lambda \frac{\bar{\ell}(w+\varepsilon, X) - \bar{\ell}(w, X)}{\varepsilon}} \right] \leq e^{\frac{\lambda^2 \xi^2}{2}}.$$

For the reverse implication, assume that $\partial_w \bar{\ell}(w, X)$ is ξ -SG for all $w \in \mathcal{W}$. Fix $w, w' \in \mathcal{W}$. By the assumptions \clubsuit we have that, \mathbb{P}_X -a.s.

$$\bar{\ell}(w', x) - \bar{\ell}(w, x) = \int_w^{w'} \partial_w \bar{\ell}(u, x) du.$$

Fix a positive integer N and let $u_j = w + j(w' - w)/N$. We have

$$\begin{aligned} \mathbb{E}_{\mathbb{P}_X} \left[e^{\lambda \frac{w' - w}{N} \sum_{j=1}^N \partial_w \bar{\ell}(u_j, X)} \right] &= \mathbb{E}_{\mathbb{P}_X} \left[\prod_{j=1}^N e^{\lambda \frac{w' - w}{N} \partial_w \bar{\ell}(u_j, X)} \right] \\ &\leq \prod_{j=1}^N \mathbb{E}_{\mathbb{P}_X} [e^{\lambda (w' - w) \partial_w \bar{\ell}(u_j, X)}]^{1/N} \leq e^{(\lambda^2 \xi^2 (w' - w)^2)/2}. \end{aligned}$$

Now let $Y_N(x) = \frac{w'-w}{N} \sum_{j=1}^N \partial_w \bar{\ell}(u_j, x)$. Since $w \mapsto \ell(w, x)$ is C^1 (\mathbb{P}_X -a.s.) by \clubsuit , we have \mathbb{P}_X -a.s. that

$$\lim_{N \rightarrow \infty} Y_N(x) = \int_w^{w'} \partial_w \bar{\ell}(u, x) du = \ell(w', x) - \ell(w, x).$$

We conclude that

$$\lim_{N \rightarrow \infty} \mathbb{E}_{\mathbb{P}_X} \left[e^{\lambda \frac{w'-w}{N} \sum_{j=1}^N \partial_w \bar{\ell}(u_j, x)} \right] = \mathbb{E}_{\mathbb{P}_X} \left[e^{\lambda(\ell(w', x) - \ell(w, x))} \right],$$

since by \clubsuit $\partial_w \bar{\ell}$ is bounded. ■

Appendix B. Extended general framework

B.1. Weakening the assumptions for the chained bounds

The framework that we presented in the main text required the assumptions \clubsuit for ℓ (or F in the setting of Theorem 21) for the chained bound. Actually a result equivalent to Theorem 4 can be obtained with weaker assumptions, namely just requiring almost sure continuity and boundedness in expectation for ℓ , and dropping the convexity hypothesis for \mathcal{W} .

Theorem 22 *Let \mathcal{W} be a compact subset of \mathbb{R}^d and $\{\mathcal{W}_k\}$ a $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} . Consider a measurable map $F : \mathcal{W} \times \mathcal{Z} \rightarrow \mathbb{R}$, such that $w \mapsto F(w, z)$ is continuous on \mathcal{W} , \mathbb{P}_Z -a.s., and $\mathbb{E}_{\mathbb{P}_Z}[\sup_{w \in \mathcal{W}} |F(w, Z)|] < +\infty$. Moreover, assume that the function $z \mapsto F(w, z) - F(w', z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi \|w - w'\|)$ wrt \mathbb{P}_Z , for every $(w, w') \in \mathcal{W}^2$. Then, we have*

$$|\mathbb{E}_{\mathbb{P}_{W,Z}}[F(W, Z)] - \mathbb{E}_{\mathbb{P}_{W \otimes Z}}[F(W, Z)]| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W_k})],$$

where $\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W_k})] = \int_{\mathcal{W}} \mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W \in \pi_k^{-1}(w)}) d\mathbb{P}_W(w)$.

Proof First notice that $w \mapsto F(w, z)$ is uniformly continuous on \mathcal{W} , \mathbb{P}_Z -a.s., since \mathcal{W} is compact. It follows that $z \mapsto \sup_{w \in \mathcal{W}} |F(w, z) - F(w_k, z)| \rightarrow 0$, \mathbb{P}_Z -a.s., and so, using the fact that this map is dominated by $z \mapsto 2 \sup_{w \in \mathcal{W}} |F(w, z)|$, which is in $L^1(\mathbb{P}_Z)$ by hypothesis, we get that

$$\mathbb{E}_{\mathbb{P}_Z} \left[\sup_{w \in \mathcal{W}} |F(w, Z) - F(w_k, Z)| \right] \rightarrow 0,$$

as $k \rightarrow +\infty$, by dominated convergence. In particular, $\mathbb{E}_{\mathbb{P}_{W,Z}}[|F(W, Z) - F(W_k, Z)|] \rightarrow 0$ and $\mathbb{E}_{\mathbb{P}_{W \otimes Z}}[|F(W, Z) - F(W_k, Z)|] \rightarrow 0$. Moreover, recalling that $\mathcal{W}_0 = \{w_0\}$ we see that $\mathbb{E}_{\mathbb{P}_{W \otimes Z}}[F(W_0, Z)] - \mathbb{E}_{\mathbb{P}_{W,Z}}[F(W_0, Z)] = 0$. It follows that

$$\begin{aligned} & |\mathbb{E}_{\mathbb{P}_{W \otimes Z}}[F(W, Z)] - \mathbb{E}_{\mathbb{P}_{W,Z}}[F(W, Z)]| \\ & \leq \sum_{k=1}^{\infty} \left| \mathbb{E}_{\mathbb{P}_{W \otimes Z}}[F(W_k, Z) - F(W_{k-1}, Z)] - \mathbb{E}_{\mathbb{P}_{W,Z}}[F(W_k, Z) - F(W_{k-1}, Z)] \right| \\ & = \sum_{k=1}^{\infty} \left| \mathbb{E}_{\mathbb{P}_{W_k \otimes Z}}[F(W_k, Z) - F(W_{k-1}, Z)] - \mathbb{E}_{\mathbb{P}_{W_k,Z}}[F(W_k, Z) - F(W_{k-1}, Z)] \right|. \end{aligned} \quad (2)$$

Now, notice that $\text{Supp}(\mathbb{P}_{Z|W_k=w_k}) \subseteq \text{Supp}(\mathbb{P}_Z)$ \mathbb{P}_W -a.s. by Lemma 30. Moreover, by the fact that $\mathbb{E}_{\mathbb{P}_Z}[\sup_{w \in \mathcal{W}} |F(w, Z)|] < +\infty$ we have $\mathbb{E}_{\mathbb{P}_{Z|W_k=w_k}}[\sup_{w \in \mathcal{W}} |F(w, Z)|] < +\infty$, and so in particular $\mathbb{E}_{\mathbb{P}_{Z|W_k=w_k}}[|F(w_k, Z) - F(w_{k-1}, Z)|] < +\infty$, for \mathbb{P}_{W_k} -almost every w_k . Thus, using the regularity of F we find that

$$\begin{aligned} & \left| \mathbb{E}_{\mathbb{P}_Z}[F(w_k, Z) - F(w_{k-1}, Z)] - \mathbb{E}_{\mathbb{P}_{Z|W_k=w_k}}[F(w_k, Z) - F(w_{k-1}, Z)] \right| \\ & \leq \xi \|w_k - w_{k-1}\| \mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W_k=w_k}), \end{aligned} \quad (3)$$

for \mathbb{P}_{W_k} -almost every w_k . We can hence conclude by taking the expectation wrt \mathbb{P}_W and using Jensen's inequality. \blacksquare

It is easy to see that the current framework includes the one in the main text.

Lemma 23 *Let $\mathcal{W} \subseteq \mathbb{R}^d$ be a convex set. Consider a measurable map $F : \mathcal{W} \times \mathcal{Z} \rightarrow \mathbb{R}$ with the following properties: $w \mapsto F(w, z)$ is C^1 \mathbb{P}_Z -a.s., $\sup_{(w,z) \in \mathcal{W} \times \mathcal{Z}} |F(w, z)| < +\infty$, and $\sup_{(w,z) \in \mathcal{W} \times \mathcal{Z}} \|\nabla_w F(w, z)\| < +\infty$. If $z \mapsto \nabla_w F(w, z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt \mathbb{P}_Z , $\forall w \in \mathcal{W}$, then $z \mapsto (F(w, z) - F(w', z))$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi \|w - w'\|)$ (wrt \mathbb{P}_Z and $\forall w, w' \in \mathcal{W}$).*

Proof Fix a probability $\hat{\mathbb{P}}_Z$ on \mathcal{Z} such that $\text{Supp}(\hat{\mathbb{P}}_Z) \subseteq \text{Supp}(\mathbb{P}_Z)$. Now, notice that since \mathcal{W} is convex, and F is C^1 , for \mathbb{P}_Z -almost every z we have

$$F(w, z) - F(w', z) = \int_0^1 \nabla_w F(w_t, z) \cdot (w - w') dt,$$

where $w_t = w' + t(w - w')$. Since F is uniformly bounded, we can use Fubini-Tonelli's theorem and Jensen's inequality to write

$$\begin{aligned} & \left| \mathbb{E}_{\mathbb{P}_Z}[F(w, Z)] - \mathbb{E}_{\hat{\mathbb{P}}_Z}[F(w', Z)] \right| \\ & \leq \int_0^1 \left| \mathbb{E}_{\mathbb{P}_Z}[\nabla_w F(w_t, Z) \cdot (w - w')] - \mathbb{E}_{\hat{\mathbb{P}}_Z}[\nabla_w F(w_t, Z) \cdot (w - w')] \right| dt. \end{aligned}$$

Using the fact that $z \mapsto F(w_t, z)$ is in both $L^1(\mathbb{P}_Z)$ and $L^1(\hat{\mathbb{P}}_Z)$ since F is bounded, we conclude by the regularity of $\nabla_w F$. \blacksquare

All the bounds of this paper can be restated in this more general framework. We will only give a direct proof of Proposition 12.

Proposition 12 *Let $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ and \mathcal{W} be a compact set, with an $\{\varepsilon_k\}$ -refining sequence of nets defined on it. Suppose that $w \mapsto \ell(w, x)$ is continuous, for \mathbb{P}_X -almost every x ,¹⁰ and that $\{\ell(w, X)\}_{w \in \mathcal{W}}$ is a ξ -SG stochastic process. Then we have*

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2I(W_k; S)}.$$

10. Note that in Asadi et al. (2018) the result is stated under a weaker assumption of separability of the process. To avoid introducing further definitions and technicalities in the proofs, we decided to focus on the case of a.s. continuity.

Proof By standard arguments, $\{\mathcal{L}_s(w)\}_{s \in \mathcal{S}}$ is a (ξ/\sqrt{m}) -SG process. Hence, $\mathcal{L}_S(w) - \mathcal{L}_S(w')$ is (ξ/\sqrt{m}) -SG for every $w, w' \in \mathcal{W}$. By Lemma 9, $s \mapsto \mathcal{L}_s(w) - \mathcal{L}_s(w')$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi/\sqrt{m})$ wrt \mathbb{P}_S ($\forall w \in \mathcal{W}$), with $\mathfrak{D} : (\mu, \nu) \mapsto \sqrt{2\text{KL}(\nu\|\mu)}$. Finally, let $g(w, s) = \mathcal{L}_X(w) - \mathcal{L}_s(w)$. Clearly g has the same regularity of \mathcal{L} . It is not hard to show that $\mathbb{E}_{\mathbb{P}_S}[\sup_{w \in \mathcal{W}} |g(w, S)|] < +\infty$ (this is a straight consequence of Remark 8.1.5 in Vershynin (2018)). We conclude by Theorem 22 and Jensen's inequality. \blacksquare

B.2. Bounds for non-uniform \mathfrak{D} -regularity

As mentioned at the end of Section 3, the results given so far are stated under uniform regularity assumptions. The next two results show that this is not strictly necessary, and that slightly different bounds can be obtained relaxing these assumptions.

Theorem 24 *Consider a non-negative measurable function $w \mapsto \xi_w$ such that $\|\xi_W\|_{L^p(\mathbb{P}_W)} = \xi$, for some $p \in [1, +\infty]$. Assume that a measurable map $F : \mathcal{W} \times \mathcal{Z} \rightarrow \mathbb{R}$ is such that $z \mapsto F(w, z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi_w)$ wrt \mathbb{P}_Z and for every $w \in \mathcal{W}$. Then we have*

$$|\mathbb{E}_{\mathbb{P}_{W \otimes Z}}[F(W, Z)] - \mathbb{E}_{\mathbb{P}_{W, Z}}[F(W, Z)]| \leq \xi \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W})^r]^{1/r},$$

where r is such that $1/p + 1/r = 1$ (with the convention $1/\infty = 0$).

Proof The proof is essentially the same as for Theorem 20, the only difference being that now we have

$$|\mathbb{E}_{\mathbb{P}_Z}[F(w, Z)] - \mathbb{E}_{\mathbb{P}_{Z|W=w}}[F(w, Z)]| \leq \xi_w \mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W=w}),$$

whose expectation under \mathbb{P}_W can be upperbounded via Hölder's inequality \blacksquare

Theorem 25 *Let \mathcal{W} be a compact subset of \mathbb{R}^d and $\{\pi_k(\mathcal{W})\}$ a $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} . Consider a measurable map $F : \mathcal{W} \times \mathcal{Z} \rightarrow \mathbb{R}$, such that $w \mapsto F(w, z)$ is continuous on \mathcal{W} , \mathbb{P}_Z -a.s., and $\mathbb{E}_{\mathbb{P}_Z}[\sup_{w \in \mathcal{W}} |F(w, Z)|] < +\infty$. Fix $\xi \geq 0$ and consider a measurable map $w \mapsto \xi_w \geq 0$ such that $\|\xi_{W_k}\|_{L^p(\mathbb{P}_W)} \leq \xi$, for all $k \in \mathbb{N}$ and for some $p \in [1, +\infty]$. Assume that for every $(w, w') \in \mathcal{W}^2$ the function $z \mapsto F(w, z) - F(w', z)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi_w \|w - w'\|)$, wrt \mathbb{P}_Z . Then, we have*

$$|\mathbb{E}_{\mathbb{P}_{W, Z}}[F(W, Z)] - \mathbb{E}_{\mathbb{P}_{W \otimes Z}}[F(W, Z)]| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W_k})^r]^{1/r},$$

where r is such that $1/p + 1/r = 1$ (with the convention $1/\infty = 0$).

Proof The proof is essentially analogous to the one of Theorem 22, but instead of (3) now we have

$$\begin{aligned} & \left| \mathbb{E}_{\mathbb{P}_{Z|W_k=w_k}}[F(w_k, Z) - F(w_{k-1}, Z)] - \mathbb{E}_{\mathbb{P}_Z}[F(w_k, Z) - F(w_{k-1}, Z)] \right| \\ & \leq \xi_{w_k} \varepsilon_{k-1} \mathfrak{D}(\mathbb{P}_Z, \mathbb{P}_{Z|W_k=w_k}). \end{aligned}$$

The conclusion follows easily by Hölder's inequality. \blacksquare

Appendix C. PAC-Bayesian bounds

The next result (Catoni, 2009) is a classical PAC-Bayesian bound. For the sake of completeness we give here a standard proof.

Proposition 16 *Assume that ℓ is bounded in $[-\xi, \xi]$. Let \mathbb{P}_W^* be a fixed probability measure on \mathcal{W} , chosen independently of S . Fix $\delta \in (0, 1)$ and $\lambda > 0$. Then, with probability $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ larger than $1 - \delta$ on the draw of S , we have*

$$\mathbb{E}_{\mathbb{P}_{W|S}}[g_S(W)] \leq \frac{\xi}{\sqrt{2m}} \left(\lambda + \frac{\text{KL}(\mathbb{P}_{W|S} \| \mathbb{P}_W^*) + \log \frac{1}{\delta}}{\lambda} \right).$$

Proof We define $\mathbb{P}_{W \otimes S}^* = \mathbb{P}_W^* \otimes \mathbb{P}_S$. Fix $\lambda > 0$. Using the Donsker-Varadhan representation of the KL divergence (Donsker and Varadhan, 1983), we have that for all $s \in \mathcal{S}$

$$\mathbb{E}_{\mathbb{P}_{W|S=s}}[g_S(W)] \leq \frac{\xi}{\sqrt{2m\lambda}} \left(\text{KL}(\mathbb{P}_{W|S=s} \| \mathbb{P}_W^*) + \log \mathbb{E}_{\mathbb{P}_W^*}[e^{\sqrt{2m}\lambda g_S(W)/\xi}] \right).$$

By Markov's inequality, we have that

$$\mathbb{P}_S \left(\mathbb{E}_{\mathbb{P}_W^*}[e^{\sqrt{2m}\lambda g_S(W)/\xi}] \leq \frac{1}{\delta} \mathbb{E}_{\mathbb{P}_{W \otimes S}^*}[e^{\sqrt{2m}\lambda g_S(W)/\xi}] \right) \geq 1 - \delta.$$

Now, for all $w \in \mathcal{W}$ we have that $\ell(w, X) \subset [-\xi, \xi]$ is ξ -SG. In particular $g_S(w)$ is (ξ/\sqrt{m}) -SG, as $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. Since $\mathbb{E}_{\mathbb{P}_S}[g_S(w)] = 0$ we have

$$\log \mathbb{E}_{\mathbb{P}_{W \otimes S}^*}[e^{\sqrt{2m}\lambda g_S(W)/\xi}] \leq \lambda^2,$$

from which we conclude. ■

Note that although Proposition 16 is valid for all $\lambda > 0$, we cannot optimise the final bound wrt λ . Indeed, we have that such a choice of λ would depend on $\text{KL}(\mathbb{P}_{W|S}, \mathbb{P}_W^*)$ and hence on the particular sample used. A possible strategy to overcome this issue consists in selecting a few possible values $\lambda_1, \dots, \lambda_t$ for λ , before drawing the sample S . Then, by mean of a union bound, one can say that with probability \mathbb{P}_S higher than $1 - t\delta$ the generalisation is bounded by the best PAC-Bayesian bound among the t ones obtained.

Proposition 17 *Under the assumptions ♣, consider a $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} and assume that $\nabla_w \ell$ is bounded in $[-\xi, \xi]$. Let \mathbb{P}_W^* be a fixed probability measure on \mathcal{W} , chosen independently of S . Fix two sequences $\{\delta_k\}_{k \in \mathbb{N}}$ and $\{\lambda_k\}_{k \in \mathbb{N}}$, such that $\delta_k \in (0, 1)$ and $\lambda_k > 0$ for all k . Assume that $\sum_{k \in \mathbb{N}} \delta_k = \delta \in (0, 1)$. Then, with probability $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$ larger than $1 - \delta$ on the draw of S , we have*

$$\mathbb{E}_{\mathbb{P}_{W|S}}[g_S(W)] \leq \frac{\xi}{\sqrt{2m}} \left(2\sqrt{\log \frac{1}{\delta_0}} + \sum_{k=1}^{\infty} \varepsilon_{k-1} \left(\lambda_k + \frac{\text{KL}(\mathbb{P}_{W_k|S} \| \mathbb{P}_{W_k}^*) + \log \frac{1}{\delta_k}}{\lambda_k} \right) \right).$$

Proof By the assumptions in ♣, $w \mapsto \mathcal{L}_s(w)$ is uniformly continuous on \mathcal{W} , \mathbb{P}_S -a.s. In particular, $\sup_{w \in \mathcal{W}} |\mathcal{L}_s(w) - \mathcal{L}_s(w_k)| \rightarrow 0$ as $k \rightarrow \infty$, \mathbb{P}_S -a.s. As a consequence $\sup_{w \in \mathcal{W}} |\mathbb{E}_{\mathbb{P}_S}[\mathcal{L}_S(w)] -$

$\mathbb{E}_{\mathbb{P}_S}[\mathcal{L}_S(w_k)] \rightarrow 0$ (\mathbb{P}_S -a.s.) since the loss is uniformly bounded. It follows that, for \mathbb{P}_S -almost every s ,

$$\lim_{k \rightarrow \infty} \mathbb{E}_{\mathbb{P}_{W|S=s}}[|g_s(W) - g_s(W_k)|] = 0.$$

Hence, recalling that $W_0 = w_0$, we have that, \mathbb{P}_S -a.s.,

$$\mathbb{E}_{\mathbb{P}_{W|S=s}}[g_s(W)] = g_s(w_0) + \sum_{k=1}^{\infty} \mathbb{E}_{\mathbb{P}_{W_k|S=s}}[g_s(W_k) - g_s(W_{k-1})].$$

On the one hand, by Hoeffding's inequality (Hoeffding, 1963), the first term in the RHS can be upper-bounded with high probability, as

$$\mathbb{P}_S \left(g_S(w_0) > \xi \sqrt{\frac{2}{m} \log \frac{1}{\delta_0}} \right) \leq \delta_0.$$

On the other hand, proceeding as in the proof of Proposition 16, for each term in the telescopic sum we can write, for \mathbb{P}_S -almost every s ,

$$\begin{aligned} & \mathbb{E}_{\mathbb{P}_{W_k|S=s}}[g_s(W_k) - g_s(W_{k-1})] \\ & \leq \frac{\varepsilon_{k-1}\xi}{\sqrt{2m\lambda_k}} \left(\text{KL}(\mathbb{P}_{W_k|S=s} \|\mathbb{P}_{W_k}^*) + \log \mathbb{E}_{\mathbb{P}_{W_k}^*} [e^{\sqrt{2m\lambda_k}(g_s(W_k) - g_s(W_{k-1}))/(\varepsilon_{k-1}\xi)}] \right). \end{aligned}$$

Now, $\nabla_w \ell \subset [-\xi, \xi]$, and hence $\nabla_w \ell(w, X)$ is ξ -SG, for all $w \in \mathcal{W}$. By Lemma 14, we have that $\{\ell(w, X)\}_{w \in \mathcal{W}}$ is a ξ -SG process. In particular, $\{g_S(w)\}_{w \in \mathcal{W}}$ is a centred (ξ/\sqrt{m}) -SG process, as $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. We have thus obtained that

$$\log \mathbb{E}_{\mathbb{P}_{W_k}^*} [e^{\sqrt{2m\lambda_k}(g_S(W_k) - g_S(W_{k-1}))/(\varepsilon_{k-1}\xi)}] \leq \lambda_k^2.$$

By Markov's inequality we have that

$$\mathbb{P}_S \left(\mathbb{E}_{\mathbb{P}_{W_k|S}}[g_s(W_k) - g_s(W_{k-1})] > \frac{\varepsilon_{k-1}\xi}{\sqrt{2m}} \left(\lambda_k + \frac{\text{KL}(\mathbb{P}_{W_k|S} \|\mathbb{P}_{W_k}^*) + \log \frac{1}{\delta_k}}{\lambda_k} \right) \right) \leq \delta_k.$$

We conclude by a union bound. ■

As for the standard PAC-Bayesian result, here as well we cannot directly optimise the parameters λ_k . Clearly one can again proceed by fixing few possible values for each parameter and then use a union argument to select the best bound. However, in this case this might become particularly hard, due to the large number of parameters. A possible way to address this problem consists in doing some optimisation that does not rely on the value of $\text{KL}(\mathbb{P}_{W_k|S}, \mathbb{P}_{W_k}^*)$, to reduce the number of parameters. For instance, we can proceed in the following way. One might suppose that $\text{KL}(\mathbb{P}_{W_k|S}, \mathbb{P}_{W_k}^*)$ increases linearly with k . Note that this is for instance the case if the algorithm is deterministic and $\mathbb{P}_{W_k}^*$ is uniform. So, let us say that we believe that $\text{KL}(\mathbb{P}_{W_k|S}, \mathbb{P}_{W_k}^*) = \alpha k$, for some $\alpha > 0$. Then we are allowed to optimise all the λ_k in the chained PAC-Bayesian bound where $\text{KL}(\mathbb{P}_{W_k|S}, \mathbb{P}_{W_k}^*)$ is replaced by αk . This leads to

$$\mathbb{E}_{\mathbb{P}_{W|S}}[g_S(W)] \leq \frac{\xi}{\sqrt{2m}} \left(2 \sqrt{\log \frac{1}{\delta_0}} + \sum_{k=1}^{\infty} \varepsilon_{k-1} \frac{\text{KL}(\mathbb{P}_{W_k|S} \|\mathbb{P}_{W_k}^*) + \log \frac{1}{\delta_k}}{\sqrt{\alpha k + \log \frac{1}{\delta_k}}} \right),$$

which is a valid bound, holding with probability higher than $1 - \delta$, for all $\alpha > 0$. Now we have essentially replaced the λ_k with a single parameter α . Again we cannot optimise directly wrt α , but we can proceed as for the unchained bound, finding a good α by means of a union bound.

As a final remark note that one might want to optimise in terms of δ_k as well. This should be possible, but the constraint $\sum_k \delta_k = \delta$ and the non-convexity of the problem can make the minimisation quite hard in practice. Yet, one can probably resort to numerical methods.

Appendix D. Omitted proofs of Section 6

Lemma 18 *Under the assumptions ♣, let ε_0 and w_0 be such that $\|w - w_0\| \leq \varepsilon_0, \forall w \in \mathcal{W}$. Assume that $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt $\mathbb{P}_S, \forall w \in \mathcal{W}$, and define $\hat{\mathcal{L}}_s(w) = \mathcal{L}_s(w) - \mathcal{L}_s(w_0)$ and $\hat{\mathcal{G}} = \mathbb{E}_{W \otimes S}[\hat{\mathcal{L}}_S(W)] - \mathbb{E}_{W,S}[\hat{\mathcal{L}}_S(W)]$. Then, $\hat{\mathcal{G}} = \mathcal{G}$, and $s \mapsto \hat{\mathcal{L}}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\varepsilon_0 \xi)$, wrt \mathbb{P}_S and $\forall w \in \mathcal{W}$.*

Proof The fact that $\mathbb{E}_{\mathbb{P}_{W,S}}[\mathcal{L}_S(w_0)] = \mathbb{E}_{\mathbb{P}_{W \otimes S}}[\mathcal{L}_S(w_0)]$ implies that $\hat{\mathcal{G}} = \mathcal{G}$. The regularity of $s \mapsto \hat{\mathcal{L}}_s(w)$ is a direct consequence of Lemma 23. ■

Lemma 19 *Assume that $\nu \mapsto \mathfrak{D}(\mathbb{P}_S, \nu)$ is convex. For any $\{\varepsilon_k\}$ -refining sequence of nets on \mathcal{W} , the sequence $\{\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})]\}_{k \in \mathbb{N}}$ is non-decreasing and, $\forall k \in \mathbb{N}$, we have*

$$\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] \leq \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W})].$$

Proof Fix $k \geq 0$ and $w_k \in \mathcal{W}_k$ such that $\mathbb{P}_W(W_k = w_k) > 0$. For any measurable set U on \mathcal{S} , we have

$$\mathbb{P}_{S|W_k=w_k}(U) = \int_{\mathcal{W}} \mathbb{P}_{S|W=w}(U) d\mathbb{P}_{W|W_k=w_k}(w),$$

where $d\mathbb{P}_{W|W_k=w_k}(w) = \frac{d\mathbb{P}_W(w)}{\mathbb{P}_W(W_k=w_k)}$ if $w \in \pi_k^{-1}(w_k)$, and 0 otherwise. Hence, we can write

$$\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k=w_k}) = \mathfrak{D}\left(\mathbb{P}_S, \int_{\mathcal{W}} \mathbb{P}_{S|W=w}(\cdot) d\mathbb{P}_{W|W_k=w_k}(w)\right).$$

Since $\nu \mapsto \mathfrak{D}(\mathbb{P}_S, \nu)$ is a convex function, we can use Jensen's inequality to obtain

$$\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k=w_k}) \leq \int_{\mathcal{W}} \mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W=w}) d\mathbb{P}_{W|W_k=w_k}(w).$$

By taking the expectation wrt \mathbb{P}_{W_k} we conclude that

$$\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] \leq \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W})].$$

Now, for any $k' > k$, the same proof can be used to show that

$$\mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] \leq \mathbb{E}_{\mathbb{P}_W}[\mathfrak{D}(\mathbb{P}_S, \mathbb{P}_{S|W_{k'}})],$$

by simply replacing \mathcal{W} with $\mathcal{W}_{k'}$ and \mathbb{P}_W with $\mathbb{P}_{W_{k'}}$. ■

Appendix E. Toy Models

E.1. Example 1

Let $\mathcal{W} = \mathcal{X} = [-1, 1]$, $\ell(w, x) = \frac{1}{2}(w - x)^2$, and $\varepsilon_k = 2^{-k}$, for $k \in \mathbb{N}$. We can find mappings π_k that define a $\{\varepsilon_k\}$ -refining sequence of nets, with $\mathcal{W}_k = \{2^{1-k}j : j \in [-2^{k-1} : 2^{k-1}]\}$, where $[a : b] = [a, b] \cap \mathbb{Z}$. Fix $k^* \in \mathbb{N}$ and define $\theta = 2^{-k^*}$. Let X be uniformly distributed on $(-\theta, \theta)$, that is $X \sim U_{(-\theta, \theta)}$. We choose an algorithm that, given x , selects the w minimising $\ell(w, x)$. This means that $\mathbb{P}_{W|X=x} = \delta_x$, where δ_x is the Dirac measure on x . Note that $\nabla_w \ell$ is 1-Lipschitz and ℓ is 2-Lipschitz (on \mathcal{X} , uniformly on \mathcal{W}). However, thanks to Lemma 18 we know that we can consider the loss $\tilde{\ell}(w, x) = \ell(w, x) - \frac{x^2}{2}$, which does not affect the algorithm, leads to the same generalisation, and is 1-Lipschitz. The marginal distribution of W is $W \sim U_{(-\theta, \theta)}$. Moreover, we have $\mathbb{E}_{\mathbb{P}_{W, X}}[\ell(W, X)] = 0$ and $\mathbb{E}_{\mathbb{P}_X}[\ell(w, X)] = \frac{1}{2} \left(w^2 + \frac{\theta^2}{3} \right)$. So,

$$\mathcal{G} = \mathbb{E}_{\mathbb{P}_{W \otimes X}}[\ell(W, X)] - \mathbb{E}_{\mathbb{P}_{W, X}}[\ell(W, X)] = \mathbb{E}_{\mathbb{P}_W} \left[\frac{1}{2} \left(W^2 + \frac{\theta^2}{3} \right) \right] = \frac{\theta^2}{3}.$$

Recall that we denote as \mathcal{B}_ℓ the bound in Proposition 11 and as $\mathcal{B}_{\nabla \ell}$ the chained bound from Proposition 15. We denote as $\mathcal{B}_{\tilde{\ell}}$ the unchained bound obtained using $\tilde{\ell}$ instead of ℓ . Clearly we have $\mathcal{B}_{\tilde{\ell}} = \mathcal{B}_\ell/2$. We will now evaluate $\mathcal{B}_{\tilde{\ell}}$ and $\mathcal{B}_{\nabla \ell}$. As a starting point, note that the 1-Wasserstein distance between two uniform measures, on the intervals (A, B) and $(a, b) \subseteq (A, B)$, is given by

$$\mathfrak{W}(U_{(A, B)}, U_{(a, b)}) = \frac{(A - a)^2 + (B - b)^2}{2((B - A) - (b - a))}.$$

Note that choosing $a = b \in [A, B]$ in the RHS above gives the 1-Wasserstein distance between a uniform distribution and a Dirac measure. Now, let $a = b = w$, $A = -\theta$ and $B = \theta$. We find that

$$\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W=w}) = \frac{\theta}{2} \left(1 + \frac{w^2}{\theta^2} \right).$$

It follows that

$$\mathcal{B}_{\tilde{\ell}} = \mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W=w})] = \frac{2}{3} \theta.$$

Comparing \mathcal{G} and $\mathcal{B}_{\tilde{\ell}}$, we realize that the standard Wasserstein bound becomes loose for small θ .

Now, fix $k \in \mathbb{N}$. If $k \leq k^*$, then $\pi_k(w) = w_0 = 0$ with probability 1. In particular, we have that $W_k \perp\!\!\!\perp X$ and hence $\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k}) = 0$. We will hence focus on the case $k > k^*$. Let $k = k^* + k'$. Now, notice that π_k defines $2^{k'} + 1$ intervals in $(-\theta, \theta)$. We will denote them as I_j , where $j \in [-2^{k'-1} : 2^{k'-1}]$. We have $I_{-2^{k'-1}} = (-\frac{1}{2^{k^*}}, -\frac{2^{k'-1}}{2^k})$ and $I_{2^{k'-1}} = (\frac{2^{k'-1}}{2^k}, \frac{1}{2^{k^*}})$, while, for $j \in [-2^{k'-1} + 1 : 2^{k'-1} - 1]$, $I_j = (\frac{2j-1}{2^k}, \frac{2j+1}{2^k})$. Note that the two outer intervals will have probability $\mathbb{P}_W(W \in I_{-2^{k'-1}}) = \mathbb{P}_W(W \in I_{2^{k'-1}}) = 2^{-(k'+1)}$, while for the inner intervals, we have $\mathbb{P}_W(W \in I_j) = 2^{-k'}$.

Now, for $j \in [-2^{k'-1} + 1 : 2^{k'-1} - 1]$ we define $a_j = \frac{2j-1}{2^k}$ and $b_j = \frac{2j+1}{2^k}$. Note that for all these inner intervals we have $b_j - a_j = 2^{1-k}$, $(b_j - a_j)/\theta = 2^{1-k'}$, $a_j/\theta = (2j-1)/2^{k'}$, and $b_j/\theta = (2j+1)/2^{k'}$. So, the contribution brought by the inner intervals to $\mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})]$

is given by

$$\begin{aligned} E_1 &= \frac{\theta}{2^{k'}} \sum_{j=-(2^{k'-1}-1)}^{2^{k'-1}-1} \frac{\left(1 + \frac{2j-1}{2^{k'}}\right)^2 + \left(-1 + \frac{2j+1}{2^{k'}}\right)^2}{4(1-2^{-k'})} \\ &= \frac{\theta}{6(1-2^{-k'})} (4 - 12 \times 2^{-k'} + 11 \times 2^{-2k'} - 3 \times 2^{-3k'}). \end{aligned}$$

On the other hand, the contribution of the two outer intervals ($j = \pm 2^{k'-1}$) is given by

$$E_2 = 2 \times \frac{\theta}{2^{k'+1}} \frac{1}{2} \frac{\left(2 - \frac{1}{2^{k'}}\right)^2}{\left(2 - \frac{1}{2^{k'}}\right)} = \frac{\theta}{2^{k'+1}} \left(2 - \frac{1}{2^{k'}}\right) = \theta \left(2^{-k'} - \frac{1}{2} \times 2^{-2k'}\right).$$

We conclude that, for $k' \geq 1$ and $k = k^* + k'$, we have

$$\begin{aligned} \mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] &= E_1 + E_2 \\ &= \frac{\theta}{6(1-2^{-k'})} (4 - 12 \times 2^{-k'} + 11 \times 2^{-2k'} - 3 \times 2^{-3k'}) + \theta \left(2^{-k'} - \frac{1}{2} \times 2^{-2k'}\right). \end{aligned}$$

We can finally compute $\mathcal{B}_{\nabla\ell}$, as we have

$$\begin{aligned} \mathcal{B}_{\nabla\ell} &= \sum_{k=1}^{\infty} \frac{1}{2^{k-1}} \mathbb{E}_{W_k}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] \\ &= \frac{1}{2^{k^*}} \sum_{k'=1}^{\infty} \frac{1}{2^{k'-1}} \mathbb{E}_{W_{k^*+k'}}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_{k^*+k'}})] = \frac{247}{105} \theta^2 \simeq 2.35 \theta^2. \end{aligned}$$

Now, it is interesting to compare these results with the CMI bound. For this purpose, we need to compute $I(W_k; X)$ for a fixed $k \in \mathbb{N}$. Similar to the chained Wasserstein bound, for $k \leq k^*$ we have that $I(W_k; X) = 0$ as $W_k \perp\!\!\!\perp X$. Therefore, we focus on $k = k^* + k'$ where $k' \geq 1$. First, notice that the KL divergence between two uniform measures, on the intervals (A, B) and $(a, b) \subseteq (A, B)$, is given by

$$\text{KL}(U_{(a,b)} \| U_{(A,B)}) = \log \frac{B-A}{b-a}.$$

As a consequence, we have that for the inner intervals I_j (with $j \in [-2^{k'-1} + 1 : 2^{k'-1} - 1]$)

$$\text{KL}(\mathbb{P}_{X|W_k \in I_j} \| \mathbb{P}_X) = \log(2^{k-k^*}) = k' \log 2,$$

while for the two outer intervals we have

$$\text{KL}(\mathbb{P}_{X|W_k \in I_{-2^{k'-1}}} \| \mathbb{P}_X) = \text{KL}(\mathbb{P}_{X|W_k \in I_{2^{k'-1}}} \| \mathbb{P}_X) = \log(2^{k+1-k^*}) = (k'+1) \log 2.$$

Taking the expectation wrt \mathbb{P}_W we obtain

$$\begin{aligned} I(W_k; X) &= \mathbb{E}_{\mathbb{P}_W}[\text{KL}(\mathbb{P}_{X|W_k} \| \mathbb{P}_X)] = \sum_{j=-2^{k'-1}}^{2^{k'-1}} \mathbb{P}_W(W_k \in I_j) \text{KL}(\mathbb{P}_{X|W_k \in I_j} \| \mathbb{P}_X) \\ &= 2 \times 2^{-(k'+1)} (k'+1) \log 2 + (1 - 2 \times 2^{-(k'+1)}) k' \log 2 = (k' + 2^{-k'}) \log 2. \end{aligned}$$

Therefore, the CMI bound is given by

$$\begin{aligned}\mathcal{B}_{\text{CMI}} &= \sum_{k=1}^{\infty} \frac{1}{2^{k-1}} \sqrt{2I(W_k; X)} \\ &= \frac{1}{2^{k^*}} \sum_{k'=1}^{\infty} \frac{1}{2^{k'-1}} \sqrt{2(k' + 2^{-k'}) \log 2} \simeq 3.50 \theta.\end{aligned}$$

For $\theta \rightarrow 0$ (i.e., $k^* \rightarrow \infty$) $\mathcal{B}_{\nabla \ell}$ is much tighter than \mathcal{B}_{ℓ} and \mathcal{B}_{CMI} , as it captures the asymptotic behaviour of $\mathcal{G} = \theta^2/3$.

Finally, let us consider the case of a random sample $S = \{X_1, \dots, X_m\}$, for $m > 1$. We denote as $\mathcal{B}_{\nabla \mathcal{L}}$ the chained Wasserstein bound, and as $\mathcal{B}_{\mathcal{L}}$ the unchained one. Minimising \mathcal{L}_S leads to

$$W = \frac{1}{m} \sum_{i=1}^m X_i.$$

Since each X_i lies in $(-\theta, \theta)$ with probability 1, in particular we have that

$$\mathbb{P}_W(W \in (-\theta, \theta)) = 1.$$

So, for $k \leq k^*$, W_k is deterministic and hence $S \perp\!\!\!\perp W_k$. We get

$$\mathcal{B}_{\nabla \mathcal{L}} = \frac{1}{\sqrt{m}} \sum_{k=1}^{\infty} \frac{1}{2^{k-1}} \mathbb{E}_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] = \frac{1}{2^{k^*} \sqrt{m}} \sum_{k'=1}^{\infty} \frac{1}{2^{k'-1}} \mathbb{E}_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] \leq 2\theta \mathcal{B}_{\mathcal{L}},$$

where we used Lemma 19 and the fact that $\theta = 2^{-k^*}$. We have thus seen that even for large samples we still have that for $\theta \rightarrow 0$

$$\frac{\mathcal{B}_{\nabla \mathcal{L}}}{\mathcal{B}_{\mathcal{L}}} = O(\theta).$$

E.1.1. HIGHER DIMENSIONAL VARIANT FOR A GENERIC LOSS

We discuss now a higher dimensional version of the above toy model. Fix a positive integer integer $d \geq 1$. Let $\mathcal{W} = \mathcal{X} = [-1, 1]^d$. Fix an integer $k^* \geq 1$ and define $\theta = 2^{-k^*}$. We will assume that the choice of k^* scales with d so that $\theta = \Theta(d^{-\alpha})$ for some $\alpha > 0$. Let X be uniformly distributed on $R_d = (\theta, \theta)^{d-1} \times (-1, 1)$. For $k \in \mathbb{N}$ we let $\varepsilon_k = 2^{-k} \sqrt{d}$ (the rescaling \sqrt{d} is necessary as now \mathcal{W} has diameter $2\sqrt{d}$) and we consider a $\{\varepsilon_k\}$ -refining sequence of nets $\mathcal{W}_k = \tilde{\mathcal{W}}_k^{\otimes d}$, where $\tilde{\mathcal{W}}_k = \{2^{1-k} j \ : \ j \in [-2^{k-1} : 2^{k-1}]\}$. We consider a generic loss function ℓ satisfying the assumptions in ♣, and such that $\nabla_w \ell$ is 1-Lipschitz in \mathcal{X} , uniformly in \mathcal{W} . From Lemma 18 we know that we can find a loss $\tilde{\ell}$ which is \sqrt{d} -Lipschitz (as $\varepsilon_0 = \sqrt{d}$), and in general we cannot assume the Lipschitz constant to be smaller. As in the 1D example, we assume that we have an algorithm that given x , selects $w = x$. This means that $\mathbb{P}_{W|X=x} = \delta_x$, where δ_x is the Dirac measure on x , and so the marginal distribution of W is U_{R_d} .

As we are interested in evaluating the Wasserstein bounds, we will need to compute quantities like $\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W=w})$ and $\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_k})$. This can be a pretty hard task if we use the standard 2-norm on \mathbb{R}^d as the distance on \mathcal{X} . To give an idea of the challenge, note that already in dimension $d = 2$ computing the expected distance between two uniform distributions on rectangles is far from

being trivial (Marsaglia et al., 1990). For this reason, everything is much easier to compute if we endow \mathcal{X} with the distance given by the 1-norm on \mathbb{R}^d , that is

$$\hat{d}_{\mathcal{X}}(x, x') = \sum_{i=1}^d |x_i - x'_i|,$$

where x_i and x'_i are the components of x and x' . We will denote the Wasserstein distances computed in this way as $\hat{\mathfrak{W}}$, and the bounds based on this distance as $\hat{\mathcal{B}}$. Note, however, that we always have that $\hat{\mathfrak{W}} \leq \mathfrak{W}$, where \mathfrak{W} is the Wasserstein distance with cost

$$d_{\mathcal{X}}(x, x') = \|x - x'\|,$$

as $d_{\mathcal{X}}(x, x') \leq \hat{d}_{\mathcal{X}}(x, x')$ for all x, x' . Moreover, when x and x' are in R_d , we have that $\hat{d}_{\mathcal{X}}(x, x') - d_{\mathcal{X}}(x, x') = O(\theta\sqrt{d-1})$. For this reason, since $\theta = \Theta(d^{-\alpha})$, we obtain that $\hat{\mathcal{B}}_{\ell} - \mathcal{B}_{\ell} = O(d^{1-\alpha})$ and $\hat{\mathcal{B}}_{\nabla\ell} - \mathcal{B}_{\nabla\ell} = O(d^{1-\alpha})$.

Now, for the Wasserstein distance between \mathbb{P}_X and $\mathbb{P}_{X|W=w}$, thanks to the fact that we are using $\hat{d}_{\mathcal{X}}$, we have

$$\hat{\mathfrak{W}}(\mathbb{P}_X, \mathbb{P}_{X|W=w}) = \sum_{i=1}^d \mathfrak{W}_{\text{1D}}(\mathbb{P}_{X_i}, \mathbb{P}_{X_i|W=w}),$$

where \mathfrak{W}_{1D} is the Wasserstein distance wrt the 1D distance $d_{\mathcal{X}_i}(x_i, x'_i) = |x_i - x'_i|$. Taking the expectation wrt \mathbb{P}_W we find

$$\hat{\mathcal{B}}_{\bar{\ell}} = \sqrt{d} \mathbb{E}_{\mathbb{P}_W}[\hat{\mathfrak{W}}(\mathbb{P}_X, \mathbb{P}_{X|W})] = \frac{2\sqrt{d}}{3} (1 - (d-1)\theta) = \Theta(d^{1/2} + d^{3/2-\alpha}).$$

Since $\hat{\mathcal{B}}_{\bar{\ell}} - \mathcal{B}_{\bar{\ell}} = O(d^{1-\alpha})$, it follows that

$$\mathcal{B}_{\bar{\ell}} = \Theta(d^{1/2} + d^{3/2-\alpha}).$$

We are now left with the task of estimating $\mathcal{B}_{\nabla\ell}$. Fix w_k such that $\mathbb{P}_W(W_k = w_k) > 0$. Now, we have that $\mathbb{P}_{X|W_k=w_k}$ is the uniform distribution on the rectangle $\pi_k^{-1}(\mathcal{W})$. Up to sets of measure 0, we can find d intervals (a_i, b_i) such that

$$\pi_k^{-1}(\mathcal{W}) = (a_1, b_1) \times \cdots \times (a_d, b_d).$$

We can choose a transport plan that is composed of d steps. First we squeeze all the probability mass from \mathcal{X} to $(a_1, b_1) \times (-1, 1)^{d-1}$. Then we squeeze the second component, and so on. In this way we find that

$$\hat{\mathfrak{W}}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_k}) \leq \sum_{i=1}^d \mathfrak{W}_{\text{1D}}(\mathbb{P}_{X_i}, \mathbb{P}_{X_i|W_k=w_k}).$$

On the other hand, we have that

$$\begin{aligned} \hat{\mathfrak{W}}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_k}) &= \inf_{\pi \in \Pi(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_k})} \mathbb{E}_{(X, X') \sim \pi} [\hat{d}_{\mathcal{X}}(X, X')] \\ &= \inf_{\pi \in \Pi(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_k})} \sum_{i=1}^d \mathbb{E}_{(X, X') \sim \pi} [|X_i - X'_i|] \geq \sum_{i=1}^d \mathfrak{W}_{\text{1D}}(\mathbb{P}_{X_i}, \mathbb{P}_{X_i|W_k=w_k}). \end{aligned}$$

We conclude that

$$\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_k}) = \sum_{i=1}^d \mathfrak{W}_{\text{ID}}(\mathbb{P}_{X_i}, \mathbb{P}_{X_i|W_k=w_k}).$$

We are now back at evaluating Wasserstein distances between uniform distributions on intervals. Proceeding as in the 1D version of the toy example we find

$$\hat{\mathcal{B}}_{\nabla\ell} = \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_S, \mathbb{P}_{S|W_k})] = \frac{247\sqrt{d}}{105} (1 + (d-1)\theta^2) = \Theta(d^{1/2} + d^{3/2-2\alpha}).$$

Again, since $\hat{\mathcal{B}}_{\nabla\ell} - \mathcal{B}_{\nabla\ell} = O(d^{1-\alpha})$ we have that if $\alpha \geq 1/2$

$$\mathcal{B}_{\nabla\ell} = \Theta(d^{1/2}).$$

In general, as α might be in $(0, 1/2)$, we can say that (since $\mathcal{B}_{\nabla\ell} \leq \hat{\mathcal{B}}_{\nabla\ell}$)

$$\mathcal{B}_{\nabla\ell} = O(d^{1/2} + d^{3/2-2\alpha}).$$

Now, we want to compare the two bounds. We have

$$\frac{\mathcal{B}_{\nabla\ell}}{\mathcal{B}_{\ell}} = O\left(\frac{1 + d^{1-2\alpha}}{1 + d^{1-\alpha}}\right).$$

If $\alpha \in (0, 1)$, we have that this ratio vanishes for $d \rightarrow \infty$, meaning that the chained bounds becomes much tighter than its unchained counterpart. On the other hand, for $\alpha > 1$ the ratio is of order 1.

E.2. Example 2

Let $\mathcal{W} = \{w \in \mathbb{R}^2 : \|w\| = 1\}$ and $\mathcal{X} = \mathbb{R}^2$. Fix $a > 0$ and let $X \sim \mathcal{N}(\mathbf{A}, \text{Id})$, a multivariate normal distribution centered in $\mathbf{A} = (a, 0)$, with covariance matrix given by the identity. Let the loss be $\ell(w, x) = -w \cdot x$. As in [Example 1](#), the algorithm selects the w minimising the loss. In practice, we are trying to find the direction of the mean of X , which is $(1, 0)$. Let $\varepsilon_k = 4/2^k$ (for $k \in \mathbb{N}$), $w_0 = (1, 0)$, and $\mathcal{W}_k = \{w = (\cos \frac{2\pi j}{2^k}, \sin \frac{2\pi j}{2^k} \phi) : j \in [-2^{k-1} : 2^{k-1} - 1]\}$ for $k \geq 1$. We can easily define projections π_k that make $\{\mathcal{W}_k\}_{k \in \mathbb{N}}$ a $\{\varepsilon_k\}$ -sequence of refining nets. With no difficulty one can verify that ℓ is 1-Lipschitz in \mathcal{X} , $\forall w \in \mathcal{W}$. Since \mathcal{W} is not convex, we want to use [Theorem 22](#) to give our chaining bound. It is easy to verify that ℓ satisfies the \mathfrak{D} regularity with $\mathfrak{D} = \mathfrak{W}$, as

$$|(\ell(w, x) - \ell(w, x')) - (\ell(w', x) - \ell(w', x'))| \leq \|x - x'\| \|w - w'\|.$$

Since the values of \mathcal{G} , \mathcal{B}_{ℓ} , and $\mathcal{B}_{\nabla\ell}$ depend on a , we will explicitly write them as functions of a . We will start by finding the exact expression of $|\mathcal{G}(a)|$.

Denote as \mathbf{a} the Cartesian axis on which \mathbf{A} lies. For $v \in \mathbb{R}^2$, denote as $\alpha(v)$ be the angle between v and \mathbf{a} . Since the learnt w is parallel to x , we have that, with probability 1, $\alpha(X) = \alpha(W)$. Thus, the distribution of $\alpha(W)$ is the distribution of the angle of an isotropic Gaussian centred in \mathbf{A} , whose density is given by ([Cooper and Farid, 2020](#))

$$\rho_a(\alpha) = \frac{\phi(a)}{\sqrt{2\pi}} \left(1 + \frac{a \cos \alpha \Phi(a \cos \alpha)}{\phi(a \cos \alpha)} \right),$$

where $\phi(t) = \frac{1}{\sqrt{2\pi}}e^{-t^2/2}$ and $\Phi(t) = \frac{1}{2}(1 + \operatorname{erf}(t/\sqrt{2}))$.

Now, we can actually give an explicit form for $|\mathcal{G}(a)|$. Indeed, we have

$$|\mathcal{G}(a)| = a \int_{-\pi}^{\pi} (1 - \cos \alpha) \rho_a(\alpha) d\alpha = a - \frac{\phi(a)}{\sqrt{2\pi}} \int_{-\pi}^{\pi} (a \cos \alpha)^2 \frac{\Phi(a \cos \alpha)}{\phi(a \cos \alpha)} d\alpha.$$

Performing a change of variable we get

$$\begin{aligned} \int_{-\pi}^{\pi} (a \cos \alpha)^2 \frac{\Phi(a \cos \alpha)}{\phi(a \cos \alpha)} d\alpha &= 2 \int_{-a}^a \frac{u^2}{\sqrt{a^2 - u^2}} \frac{\Phi(u)}{\phi(u)} du \\ &= \frac{a^2 e^{a^2/4} \pi^{3/2}}{\sqrt{2}} (I_0(a^2/a) + I_1(a^2/4)), \end{aligned}$$

where $I_n(t)$ denotes the modified Bessel function of the first kind. So, we have

$$|\mathcal{G}(a)| = a \left(1 - \frac{1}{2} \frac{I_0(a^2/a) + I_1(a^2/4)}{\sqrt{\frac{2}{\pi} \frac{e^{a^2/4}}{a}}} \right).$$

We can now use the asymptotic expansions

$$\begin{aligned} I_0(a^2/4) &= \sqrt{\frac{2}{\pi}} \frac{e^{a^2/4}}{a} \left(1 + \frac{1}{2a^2} + O(a^{-4}) \right); \\ I_1(a^2/4) &= \sqrt{\frac{2}{\pi}} \frac{e^{a^2/4}}{a} \left(1 - \frac{3}{2a^2} + O(a^{-4}) \right), \end{aligned}$$

to get that

$$|\mathcal{G}(a)| = \frac{1}{2a} + O(a^{-3}).$$

Now, we want to show that, as $a \rightarrow \infty$, \mathcal{B}_ℓ is of order 1. We start by computing a lower bound. For each w , let us consider a new set of Cartesian axes ($\mathbf{u}(w)$ and $\mathbf{v}(w)$), such that the angle between $\mathbf{v}(w)$ and \mathbf{a} is $\alpha(w)$, and $\mathbf{u}(w)$ is the normal axis which contains the point \mathbf{A} . We choose the orientation of the axes so that in this reference framework we have $\mathbf{A} = (a \sin \alpha(w), 0)$. Since X , conditioned on $\mathcal{W} = w$, has support contained in the axis $\mathbf{v}(w)$, the Wasserstein distance $\mathcal{W}(\mathbb{P}_X, \mathbb{P}_{X|W=w})$ is lower-bounded by the transport cost of moving every point in \mathbb{R}^2 to the closest point on $\mathbf{v}(w)$. We thus have

$$\begin{aligned} \mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W=w}) &\geq \frac{1}{2\pi} \int_{\mathbb{R}^2} |u| e^{-\frac{(u-a \sin \alpha(w))^2 + v^2}{2}} du dv \\ &= \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} |u| e^{-\frac{(u-a \sin \alpha(w))^2}{2}} du \geq a |\sin \alpha(w)|. \end{aligned}$$

We can now explicitly compute a lower bound for $\mathcal{B}_\ell(a)$ by taking the expectation wrt \mathbb{P}_W . We get

$$\mathcal{B}_\ell(a) \geq \int_{-\pi}^{\pi} a |\sin \theta| \rho_a(\theta) d\theta = \sqrt{\frac{2}{\pi}} \operatorname{erf} \frac{a}{\sqrt{2}}.$$

In particular, we have established that

$$\liminf_{a \rightarrow \infty} \mathcal{B}_\ell(a) \geq \sqrt{\frac{2}{\pi}}.$$

We can now look for an upper bound on $\mathcal{B}_\ell(a)$. Fixed w , we can consider the following transport plan from \mathbb{P}_X to $\mathbb{P}_{X|W=w}$. First, we transport all the probability mass on $\mathfrak{v}(w)$, then we arrange the mass on $\mathfrak{v}(w)$ so as to reach the correct density. For the first step, notice that we are simply projecting \mathbb{P}_X on $\mathfrak{v}(w)$. It is not hard to realise that in this way the linear density obtained on $\mathfrak{v}(w)$ is a centred standard normal distribution. The transport cost for this step is given by

$$\frac{1}{2\pi} \int_{\mathbb{R}^2} |u| e^{-\frac{(u-a \sin \alpha(w))^2 + v^2}{2}} du dv \leq 1 + a |\sin \alpha(w)|.$$

Now let $V \sim \mathcal{N}(0, 1)$. The actual distribution of $\mathbb{P}_{X|W=w}$ on $\mathfrak{v}(w)$ is actually given by V , conditioned on $V \geq -a \cos \alpha(w)$, as $-a \cos \alpha(w)$ is the coordinate on $\mathfrak{v}(w)$ of the origin of the standard \mathbb{R}^2 Cartesian framework and so $\mathbb{P}_{X|W=w}$ has support $\{v \in \mathfrak{v}(w) : v \geq -a \cos \alpha(w)\}$. We can easily evaluate

$$\mathfrak{W}(\mathbb{P}_V, \mathbb{P}_{V|V \geq -a \cos \alpha(w)}) = \frac{\phi(a \cos \alpha(w))}{\Phi(a \cos \alpha(w))}.$$

So we have found that

$$\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W=w}) \leq 1 + a |\sin \alpha(w)| + \frac{\phi(a \cos \alpha(w))}{\Phi(a \cos \alpha(w))}.$$

Averaging on w we get that

$$\mathcal{B}_\ell(a) = \mathbb{E}_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W})] \leq 1 + \sqrt{\frac{2}{\pi}} \operatorname{erf} \frac{a}{\sqrt{2}} + \frac{e^{-a^2}}{\Phi(-a)},$$

and so

$$\limsup_{a \rightarrow \infty} \mathcal{B}_\ell(a) \leq 1 + \sqrt{\frac{2}{\pi}}.$$

In particular, we have found that $\mathcal{B}_\ell(a) = \Theta(1)$, for $a \rightarrow \infty$.

We are now left with the task of evaluating $\mathcal{B}_{\nabla \ell}(a)$. Recall that, for each $k \geq 1$, we have $\mathcal{W}_k = \{w = (\cos \frac{2\pi j}{2^k}, \sin \frac{2\pi j}{2^k}) : j \in [-2^{k-1} : 2^{k-1} - 1]\}$ and $w_0 = (1, 0)$. Denote as \mathcal{U}_k the partition on \mathcal{W} induced by π_k , that is

$$\mathcal{U}_k = \{U = \pi_k^{-1}(w) : w \in \mathcal{W}_k\}.$$

We can certainly suppose that each $U \in \mathcal{U}_k$ is the circular arc enclosed by two adjacent elements of \mathcal{W}_k . Now, let $\bar{\mathcal{U}}_k = \{U \in \mathcal{U}_k : (1, 0) \neq U\}$ and define $\theta_k = \pi/2^k$. Then, we have that, up to points of null measure, $\{W_k = (1, 0)\} = \{|\alpha(W)| \leq \theta_k\}$. As a consequence

$$\begin{aligned} \mathbb{E}_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] &= \sum_{U \in \mathcal{U}_k} \mathbb{E}_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W \in U}) \mathbf{1}_U(W)] \\ &= \mathbb{P}_W(|\alpha(W)| \leq \theta_k) \mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_0}) + \sum_{U \in \bar{\mathcal{U}}_k} \mathbb{P}_W(W \in U) \mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W \in U}), \end{aligned} \quad (4)$$

where $\mathbf{1}_U$ is the indicator function of the event U . We need now to upper-bound the terms of this sum.

Let us define $Z = X - \mathbf{A}$. Clearly $Z \sim \mathcal{N}(0, \text{Id})$. Let ρ be the density of Z , a centered standard multivariate normal, and $\tilde{\rho}$ be the density of Z conditioned on $|\alpha(W)| \leq \theta_k$. We have that

$$\tilde{\rho}(z) = \begin{cases} 0 & \text{if } |\alpha| > \theta; \\ \rho(z)/\mathbb{P}_W(|\alpha(W)| \leq \theta_k) & \text{otherwise.} \end{cases}$$

Let $\zeta = \|Z\|$ and note that $\zeta \sim \chi_2$, the Rayleigh distribution. We notice that

$$\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_0}) = \mathfrak{W}(\mathbb{P}_Z, \mathbb{P}_{Z||\alpha(W)| \leq \theta_k}).$$

We can upper-bound this quantity by the transport cost of moving the mass $\mathbb{P}_W(|\alpha(W)| > \theta_k)$ away from $\{|\alpha(W)| > \theta_k\}$, bringing it all on \mathbf{A} , and finally redistributing it in the slice $\{|\alpha(W)| \leq \theta_k\}$, proportionally to $\tilde{\rho}$. We hence have

$$\mathfrak{W}(\mathbb{P}_Z, \mathbb{P}_{Z||\alpha(W)| \leq \theta_k}) \leq \mathbb{P}_W(|\alpha(W)| > \theta_k) (\mathfrak{W}(\mathbb{P}_Z, \delta_{\mathbf{A}}) + \mathfrak{W}(\delta_{\mathbf{A}}, \mathbb{P}_{Z||\alpha(W)| \leq \theta_k})).$$

We can evaluate

$$\mathfrak{W}(\mathbb{P}_Z, \delta_{\mathbf{A}}) = \int_{\mathbb{R}^2} \|z\| \rho(z) dz = \mathbb{E}_{\zeta}[\zeta] = \sqrt{\frac{\pi}{2}}.$$

On the other hand,

$$\begin{aligned} \mathfrak{W}(\delta_{\mathbf{A}}, \mathbb{P}_{Z||\alpha(W)| \leq \theta_k}) &= \int_{\mathbb{R}^2} \|z\| \tilde{\rho}(z) dz \\ &\leq \frac{1}{\mathbb{P}_W(|\alpha(W)| \leq \theta_k)} \int_{\mathbb{R}^2} \|z\| \rho(z) dz = \frac{\sqrt{\pi/2}}{\mathbb{P}_W(|\alpha(W)| \leq \theta_k)}. \end{aligned}$$

Now notice that

$$\mathbb{P}_W(|\alpha(W)| \leq \theta_k) \geq \mathbb{P}_{\zeta}(\zeta \leq a \sin \theta_k) = F_{\zeta}(a \sin \theta_k),$$

where $F_{\zeta} : u \mapsto 1 - e^{-u^2/2}$ is the cdf of ζ . As a consequence we eventually find

$$\begin{aligned} \mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k=w_0}) &\leq \mathbb{P}_W(|\alpha(W)| > \theta_k) \left(1 + \frac{1}{\mathbb{P}_W(|\alpha(W)| \leq \theta_k)} \right) \sqrt{\frac{\pi}{2}} \leq \left(1 + \frac{1}{F_{\zeta}(a \sin \theta_k)} \right) \sqrt{\frac{\pi}{2}}. \end{aligned}$$

Now, for $U \in \bar{U}_k$, we have that $\{W \in U\} \subseteq \{|\alpha(W)| \geq \theta_k\}$. We can upper-bound $\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W \in U}) = \mathfrak{W}(\mathbb{P}_Z, \mathbb{P}_{Z|W \in U})$ as

$$\mathfrak{W}(\mathbb{P}_Z, \mathbb{P}_{Z|W \in U}) \leq \mathfrak{W}(\mathbb{P}_Z, \delta_{\mathbf{A}}) + \mathfrak{W}(\delta_{\mathbf{A}}, \mathbb{P}_{Z|W \in U}).$$

We have already computed $\mathfrak{W}(\mathbb{P}_Z, \delta_{\mathbf{A}}) = \sqrt{\pi/2}$. For the other term we have

$$\begin{aligned} \mathfrak{W}(\delta_{\mathbf{A}}, \mathbb{P}_{Z|W \in U}) &= \frac{1}{\mathbb{P}_W(W \in U)} \int_{(z+\mathbf{A})/\|z+\mathbf{A}\| \in U} \|z\| \rho(z) dz \\ &\leq \frac{1}{\mathbb{P}_W(W \in U)} \int_{\|z\| > a \sin \theta_k} \|z\| \rho(z) dz = \frac{1 - F_{\zeta}(a \sin \theta_k)}{\mathbb{P}(W \in U)}. \end{aligned}$$

We have thus obtained that

$$\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W \in U}) \leq \sqrt{\frac{\pi}{2}} + \frac{1 - F_\zeta(a \sin \theta_k)}{\mathbb{P}(W \in U)}.$$

Going back to (4), we can now write

$$\mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] \leq (1 - F_\zeta(a \sin \theta_k)) \left(\left(2 + \frac{1}{F_\zeta(a \sin \theta_k)} \right) \sqrt{\frac{\pi}{2}} + 2^k - 1 \right), \quad (5)$$

where we used that \bar{U}_k has $2^k - 1$ elements and that $\sum_{u \in \bar{U}_k} \mathbb{P}_W(W \in U) \leq (1 - F_\zeta(a \sin \theta_k))$. Now, by plugging into (5) the explicit expressions of F_ζ and θ_k we obtain

$$\mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] \leq e^{-\frac{1}{2}a^2 \sin^2(\pi/2^k)} \left(2^k - 1 + \left(2 + \frac{1}{1 - e^{-\frac{1}{2}a^2 \sin^2(\pi/2^k)}} \right) \sqrt{\frac{\pi}{2}} \right) = \mathcal{B}_k(a).$$

Fix $k^* > 1$. By Lemma 19, we have that for all $k \leq k^*$, $\mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] \leq \mathcal{B}_{k^*}(a)$, and for $k > k^*$ we have $\mathbb{E}_{\mathbb{P}_W}[\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X|W_k})] \leq \mathcal{B}_\ell(a)$. So we have that

$$\mathcal{B}_{\nabla \ell}(a) \leq \sum_{k=1}^{k^*} \varepsilon_{k-1} \mathcal{B}_{k^*}(a) + \sum_{k=k^*}^{\infty} \varepsilon_k \mathcal{B}_\ell(a) \leq 8 \mathcal{B}_{k^*}(a) + 4 \times 2^{-k^*} \mathcal{B}_\ell(a).$$

Now the idea is that we want to choose $k^* = k_a^*$ as a function of a , in a way that makes the bound vanish for $a \rightarrow +\infty$. Note that if

$$a \geq \frac{2 \log 2 \sqrt{k_a^*}}{\sin(\pi/2^{k_a^*})}, \quad (6)$$

then

$$\mathcal{B}_{k_a^*}(a) \leq 2^{-k_a^*} + 2^{-2k_a^*} \left(2 + \frac{1}{1 - 2^{-2k_a^*}} \right) \sqrt{\frac{\pi}{2}}.$$

Notice we can choose $a \mapsto k_a^*$ such that (6) holds and $a = O(2^{-k_a^*} \sqrt{k_a^*})$, for $a \rightarrow +\infty$, which implies

$$2^{-k_a^*} = O\left(\frac{\log a - \log \log a}{a}\right).$$

This proves the asymptotic behaviour for large a

$$\mathcal{B}_{\nabla \ell}(a) = O\left(\frac{\log a - \log \log a}{a}\right).$$

In particular, up to logarithmic factors, the chained bound can capture the correct behaviour of $\mathcal{G}(a)$.

Appendix F. Technicalities

Lemma 26 *The mapping $w \mapsto \mathbb{P}_{Z|W=w}$ is measurable.*

Proof Recall that $\Sigma_{\mathcal{P}}$ is the σ -algebra on \mathcal{P} induced by the weak topology. $\Sigma_{\mathcal{P}}$ is generated by the maps $\phi_U : \mathcal{P} \rightarrow [0, 1]$, given by $\mu \mapsto \phi_U(\mu) = \mu(U)$, for U ranging in $\Sigma_{\mathcal{Z}}$ (cf. Theorem 17.24 in [Kechris \(1995\)](#)). By definition of regular conditional probability, for every $U \in \Sigma_{\mathcal{Z}}$ the map $w \mapsto \mathbb{P}_{Z|W=w}(U)$ is measurable. Hence $w \mapsto \mathbb{P}_{Z|W=w}$ is a measurable map $\mathcal{W} \rightarrow \mathcal{P}$ wrt $\Sigma_{\mathcal{P}}$. ■

Definition 27 Let $f : (0, +\infty) \rightarrow \mathbb{R}$ be a convex lower semi-continuous map such that $f(1) = 0$ and $\lim_{x \rightarrow +\infty} f(x)/x = +\infty$. For $\mu, \nu \in \mathcal{P}$ we define the f -divergence

$$D_f(\nu \parallel \mu) = \begin{cases} \mathbb{E}_{\mu}[f(\frac{d\nu}{d\mu})] & \text{if } \nu \ll \mu; \\ +\infty & \text{otherwise.} \end{cases}$$

Examples of f divergences are the KL divergence ($f : u \mapsto u \log u$) and the p -power divergence ($f : u \mapsto u^p - 1$).

Lemma 28 $\mathfrak{D} : \mathcal{P} \times \mathcal{P} \rightarrow [0, +\infty]$, defined by $\mathfrak{D}(\mu, \nu) = D_f(\nu \parallel \mu)$, is measurable.

Proof The measurability follows from the fact D_f is weakly lower semi-continuous (see Corollary 2.9 and Remark 2.1 in [Liero et al. \(2018\)](#)). ■

Lemma 29 $\mathfrak{D} : \mathcal{P} \times \mathcal{P} \rightarrow [0, +\infty]$, defined by $\mathfrak{D}(\mu, \nu) = \mathfrak{W}(\mu, \nu)$, is measurable.

Proof The measurability follows from the weak lower semi-continuity of \mathfrak{W} (see [Villani \(2009\)](#), Remark 6.12). ■

Lemma 30 $\text{Supp}(\mathbb{P}_{Z|W=w}) \subseteq \text{Supp}(\mathbb{P}_Z)$, \mathbb{P}_W -a.s.

Proof We start by recalling that given a measure $\mu \in \mathcal{P}$, $\text{Supp}(\mu)$ is the smallest closed subset K of \mathcal{Z} such that $\mu(K) = 1$. Let $U \subseteq \mathcal{W}$ be defined as

$$U = \{w \in \mathcal{W} : \mathbb{P}_{Z|W=w}(\text{Supp}(\mathbb{P}_Z)) < 1\}.$$

First, we notice that U is measurable. Indeed, $\text{Supp}(\mathbb{P}_Z)$ is closed, and hence measurable, so $w \mapsto \mathbb{P}_{Z|W=w}(\text{Supp}(\mathbb{P}_Z))$ is a measurable map, by definition of regular conditional probability. Now note that

$$\begin{aligned} 1 = \mathbb{P}_Z(\text{Supp}(\mathbb{P}_Z)) &= \int_{\mathcal{W}} \mathbb{P}_{Z|W=w}(\text{Supp}(\mathbb{P}_Z)) \, d\mathbb{P}_W(w) \\ &\leq 1 - \mathbb{P}_W(U) + \int_U \mathbb{P}_{Z|W=w}(\text{Supp}(\mathbb{P}_Z)) \, d\mathbb{P}_W(w). \end{aligned}$$

As a consequence, we must have that $\int_U \mathbb{P}_{Z|W=w}(\text{Supp}(\mathbb{P}_Z)) \, d\mathbb{P}_W(w) \geq \mathbb{P}_W(U)$. However, by definition $\mathbb{P}_{Z|W=w}(\text{Supp}(\mathbb{P}_Z)) < 1$ for $w \in U$, and so we necessarily have $\mathbb{P}_W(U) = 0$. We conclude by noticing that $\text{Supp}(\mathbb{P}_{Z|W=w}) \supset \text{Supp}(\mathbb{P}_Z)$ if, and only if, $w \in U$. ■

Appendix G. Explicit bounds

In this section we present several bounds that can be derived via the framework of Section 3. To our knowledge, all the chaining bounds that we present here are new, the only exception being the one in Proposition 43, which was recently established in Zhou et al. (2022). However, most of the unchained counterparts were already derived in the literature. The reader can find the bibliographic references in Table 1. Henceforth, all the chained bounds that we state are valid for any $\{\varepsilon_k\}$ -sequence of refining nets on \mathcal{W} .

G.1. A few examples of \mathfrak{D} -regularity

Definition 31 (Power divergence) Let $p > 1$. Given two probabilities μ and ν on \mathcal{Z} , we define the p -power divergence

$$D^{(p)}(\nu\|\mu) = \begin{cases} \mathbb{E}_\mu \left[\left(\frac{d\nu}{d\mu} \right)^p \right] - 1 & \text{if } \nu \ll \mu; \\ +\infty & \text{otherwise.} \end{cases}$$

For $p = 2$, we denote $D^{(2)}(\nu\|\mu)$ as $\chi^2(\nu\|\mu)$.

Lemma 32 Fix $p > 1$ and let $r = p/(p-1)$. Let $\mathfrak{D} : (\mu, \nu) \mapsto (D^{(p)}(\nu\|\mu) + 1)^{1/p}$ and $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ be measurable. Assume that $f \in L^1(\mu)$ and write $f_\mu = \mathbb{E}_\mu[f(Z)]$. If $\mathbb{E}_\mu[\|f(Z) - f_\mu\|^r]^{1/r} \leq \xi$, then f has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt μ .

Proof Notice that \mathfrak{D} is measurable by Lemma 28. First, we consider the case $q = 1$. Fix $\nu \in \mathcal{P}$ such that $\text{Supp}(\nu) \subseteq \text{Supp}(\mu)$ and $f \in L^1(\nu)$. If ν is not absolutely continuous wrt μ , than the claim is trivially true, so assume $\nu \ll \mu$. Define $f_\mu = \mathbb{E}_\mu[f(Z)]$. We have

$$\begin{aligned} |\mathbb{E}_\mu[f(Z)] - \mathbb{E}_\nu[f(Z)]| &\leq \mathbb{E}_\nu[|f(Z) - f_\mu|] = \int_{\mathcal{Z}} |f(z) - f_\mu| \frac{d\nu}{d\mu}(z) d\mu(z) \\ &\leq \mathbb{E}_\mu[|f(Z) - f_\mu|^r]^{1/r} (D^{(p)}(\nu\|\mu) + 1)^{1/p}, \end{aligned}$$

by Hölder's inequality.

The case $q > 1$ follows from the one-dimensional case, since $\mathbb{E}_\mu[|(f(Z) - f_\mu) \cdot v|^r]^{1/r} \leq \|v\| \mathbb{E}_\mu[\|(f(Z) - f_\mu)\|^r]^{1/r}$ for all $v \in \mathbb{R}^q$. \blacksquare

Corollary 33 Fix $p > 1$ and let $r = p/(p-1)$. Let $\mathfrak{D} : (\mu, \nu) \mapsto (D^{(p)}(\nu\|\mu) + 1)^{1/p}$ and $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ be measurable. Assume that $f(Z)$ is ξ -SG for $Z \sim \mu$. Then f has regularity $\mathcal{R}_{\mathfrak{D}}(e^{1/e} \sqrt{r} \xi)$ wrt μ .

Proof Simply use that $\mathbb{E}_\mu[\|f(Z) - \mathbb{E}_{Z' \sim \mu}[f(Z')]\|^r]^{1/r} \leq e^{1/e} \sqrt{r} \xi$ if $f(Z)$ is ξ -SG to conclude by Lemma 32. \blacksquare

Lemma 34 Let $\mathfrak{D} : (\mu, \nu) \mapsto \sqrt{\chi^2(\nu\|\mu)}$. Let $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ be measurable. Assume that $\|\mathbb{C}_\mu[f(Z)]\| \leq \xi^2$, where $\mathbb{C}_\mu[f(Z)]$ is the covariance matrix of $f(Z)$ for $Z \sim \mu$. Then, f has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$.

Proof For $q = 1$, the claim is a direct consequence of the HCR bound (Lehmann and Casella, 1998). The case $q > 1$ follows easily. ■

Definition 35 (Total variation) *The total variation of two probability measures $\mu, \nu \in \mathcal{P}$ is defined as*

$$\text{TV}(\mu, \nu) = \sup_{U \in \Sigma_{\mathcal{Z}}} |\mu(U) - \nu(U)|.$$

Lemma 36 *Let $\mathfrak{D} : (\mu, \nu) \mapsto 2\text{TV}(\mu, \nu)$. Let $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ be a measurable map, bounded in $[-\xi, \xi]$. Then f has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$ wrt any $\mu \in \mathcal{P}$.*

Proof First, we need to show that $\nu \mapsto \text{TV}(\mu, \nu)$ is measurable. We have that for all $U \in \Sigma_{\mathcal{Z}}$, the map $\nu \mapsto |\mu(U) - \nu(U)|$ is continuous in the weak topology. In particular, taking the supremum wrt U we get a weakly lower semicontinuous map, which implies the measurability. Now, notice that asking $f \subseteq [-\xi, \xi]$ is equivalent to ask for f to be 2ξ -Lipschitz wrt the discrete metric on \mathcal{Z} . We can then proceed as in Lemma 9 using the fact that the total variation coincides with the 1-Wasserstein distance when the transport cost is the discrete metric (Villani, 2009). ■

Corollary 37 *Let $\mathfrak{D} : (\mu, \nu) \mapsto \sqrt{2\text{KL}(\mu\|\nu)}$. Let $f : \mathcal{Z} \rightarrow \mathbb{R}^q$ be a measurable map, bounded in $[-\xi, \xi]$. Then f has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$.*

Proof The measurability of \mathfrak{D} is a obvious consequence of Lemma 28. Then, the claim follows directly from Lemma 36 by Pinsker's inequality; see e.g. van Handel (2016). ■

G.2. Some simple bounds based on the \mathfrak{D} -regularity

Definition 38 (Power information) *Consider two coupled random variables Z, Z' on $(\mathcal{Z}, \Sigma_{\mathcal{Z}})$. For $p > 1$ we define their p -power information (Guntuboyina et al., 2014) as*

$$I^{(p)}(Z; Z') = D^{(p)}(\mathbb{P}_{Z, Z'} \|\mathbb{P}_{Z \otimes Z'}).$$

Proposition 39 *Fix $p > 1$, let $r = p/(p-1)$ and suppose that $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. On the one hand, if $\ell(w, X)$ is ξ -SG for $X \sim \mathbb{P}_X$, for all $w \in \mathcal{W}$, then*

$$|\mathcal{G}| \leq \frac{e^{1/e} \sqrt{r} \xi}{\sqrt{m}} (I^{(p)}(S; W) + 1)^{1/p}.$$

On the other hand, under the assumptions ♣ if $\nabla_w \ell$ is ξ -SG for $X \sim \mathbb{P}_X$, for all $w \in \mathcal{W}$, then

$$|\mathcal{G}| \leq \frac{e^{1/e} \sqrt{r} \xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} (I^{(p)}(S; W_k) + 1)^{1/p}.$$

Proof First notice that the ξ -subgaussianity of ℓ (respectively $\nabla_w \ell$) implies that of \mathcal{L} (respectively $\nabla_w \mathcal{L}$) is (ξ/\sqrt{m}) -SG. Then, the first claim follows by Corollary 33, Theorem 2, and Jensen's inequality, while the second one by Corollary 33, Theorem 4, and Jensen's inequality. ■

Proposition 40 Suppose that $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. On the one hand, if $\mathbb{V}_{\mathbb{P}_X}[\ell(w, X)] \leq \xi^2$, for all $w \in \mathcal{W}$, then

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \mathbb{E}_{\mathbb{P}_W} \left[\sqrt{\chi^2(\mathbb{P}_{S|W} \|\mathbb{P}_S)} \right].$$

On the other hand, under the assumptions ♣ if $\|\mathbb{C}_{\mathbb{P}_X}[\nabla_w \ell(w, X)]\| \leq \xi^2$, for all $w \in \mathcal{W}$, then

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} \left[\sqrt{\chi^2(\mathbb{P}_{S|W_k} \|\mathbb{P}_S)} \right].$$

Proof The claims follow combining Lemma 34 with Theorems 2 and 4. Note that the variance of \mathcal{L} is re-scaled by a factor $1/\sqrt{m}$ wrt the one of ℓ , as $\mathbb{P}_S = \mathbb{P}_X^{\otimes m}$. The same is true for the covariance of $\nabla_w \mathcal{L}$. ■

G.3. Individual-sample bounds

Recall that $S = \{X_1, \dots, X_m\}$. In this section we will consider a probability measure \mathbb{P}_S on (\mathcal{S}, Σ_S) such that the marginals $\mathbb{P}_{X_i} = \mathbb{P}_X$ for all $i \in [1 : m]$, but we do not require that the draws are independent. Note moreover that W might depend in a different way on each X_i , so that we can have that $\mathbb{P}_{W, X_i} \neq \mathbb{P}_{W, X_j}$, for $i \neq j$. Now, we specialise Theorems 2 and 4 to obtain individual-sample bounds, such as those from Bu et al. (2019).

Proposition 41 Assume that $x \mapsto \ell(w, x)$ has regularity $\mathcal{R}_{\mathcal{D}}(\xi)$ wrt \mathbb{P}_X , $\forall w \in \mathcal{W}$. Then we have

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_W} [\mathcal{D}(\mathbb{P}_X, \mathbb{P}_{X_i|W})].$$

Proof Just write

$$\mathcal{G} = \frac{1}{m} \sum_{i=1}^m (\mathbb{E}_{\mathbb{P}_{W \otimes X}} [\ell(W, X)] - \mathbb{E}_{\mathbb{P}_{W, X_i}} [\ell(W, X_i)]).$$

and then conclude by applying Theorem 20 to bound each term of the sum. ■

Proposition 42 Assume ♣ and suppose that $x \mapsto \ell(w, x)$ has regularity $\mathcal{R}_{\mathcal{D}}(\xi)$ wrt \mathbb{P}_X , $\forall w \in \mathcal{W}$. Then we have

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\mathcal{D}(\mathbb{P}_X, \mathbb{P}_{X_i|W_k})].$$

Proof Just write

$$\mathcal{G} = \frac{1}{m} \sum_{i=1}^m (\mathbb{E}_{\mathbb{P}_{W \otimes X}} [\ell(W, X)] - \mathbb{E}_{\mathbb{P}_{W, X_i}} [\ell(W, X_i)]).$$

and then conclude by applying Theorem 21 to bound each term of the sum. ■

We can now state several individual-sample generalisation bounds. For the sake of brevity, we will omit the proofs, as they are all direct applications of Propositions 41 and 42, and of the previously established results of \mathcal{D} -regularity.

Proposition 43 *On the one hand, if $\ell(w, X)$ is ξ -SG uniformly on \mathcal{W} , then*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sqrt{2I(W; X_i)}.$$

On the other hand, if $\nabla_w \ell(w, X)$ is ξ -SG uniformly on \mathcal{W} , then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2I(W_k; X_i)}.$$

Proposition 44 *On the one hand, if $x \mapsto \ell(w, x)$ is ξ -Lipschitz uniformly on \mathcal{W} , then*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m E_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X_i|W})].$$

On the other hand, assume ♣. If $x \mapsto \nabla_w \ell(w, x)$ is ξ -Lipschitz uniformly on \mathcal{W} , then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} E_{\mathbb{P}_W} [\mathfrak{W}(\mathbb{P}_X, \mathbb{P}_{X_i|W_k})].$$

Proposition 45 *Fix $p > 1$ and let $r = p/(p-1)$. Write $\bar{\ell}(w)$ for $\mathbb{E}_{\mathbb{P}_X} [\ell(w, X)]$ and $\overline{\nabla_w \ell}(w)$ for $\mathbb{E}_{\mathbb{P}_X} [\nabla_w \ell(w, X)]$. On the one hand, if, for all $w \in \mathcal{W}$, $\mathbb{E}_{\mathbb{P}_X} [|\ell(w, X) - \bar{\ell}(w)|^r] \leq \xi^r$, then*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m (I^{(p)}(W; X_i) + 1)^{1/p}.$$

On the other hand, assume ♣. If $\mathbb{E}_{\mathbb{P}_X} [\|\nabla_w \ell(w, X) - \overline{\nabla_w \ell}(w)\|^r] \leq \xi^r$, for all $w \in \mathcal{W}$, then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} (I^{(p)}(W_k; X_i) + 1)^{1/p}.$$

Proposition 46 *On the one hand, if, for all $w \in \mathcal{W}$, $\mathbb{V}_{\mathbb{P}_X} [\ell(w, X)] \leq \xi^2$, then*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_W} \left[\sqrt{\chi^2(\mathbb{P}_{X_i|W} \|\mathbb{P}_X)} \right].$$

On the other hand, assume ♣. If $\|\mathbb{C}_{\mathbb{P}_X} [\nabla_w \ell(w, X)]\| \leq \xi^2$, for all $w \in \mathcal{W}$, then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} \left[\sqrt{\chi^2(\mathbb{P}_{X_i|W_k} \|\mathbb{P}_X)} \right].$$

Proposition 47 *On the one hand, if $|\ell(w, x)| \leq \xi$ for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$, then*

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_W} [\mathfrak{TV}(\mathbb{P}_X, \mathbb{P}_{X_i|W})].$$

On the other hand, assume ♣. If $\|\nabla_w \ell(w, x)\| \leq \xi$ for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$, then

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\mathfrak{TV}(\mathbb{P}_X, \mathbb{P}_{X_i|W_k})].$$

Definition 48 (Lautum information) Consider two coupled random variables Z, Z' on $(\mathcal{Z}, \Sigma_{\mathcal{Z}})$. We define their lautum information (Palomar and Verdú, 2008) as

$$L(Z; Z') = \text{KL}(\mathbb{P}_{Z \otimes Z'} \| \mathbb{P}_{Z, Z'}).$$

Proposition 49 On the one hand, if $|\ell(w, x)| \leq \xi$ for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$, then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sqrt{2L(W; X_i)}.$$

On the other hand, assume ♣. If $\|\nabla_w \ell(w, x)\| \leq \xi$ for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$, then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2L(W_k; X_i)}.$$

G.4. Bounds based on random sub-sampling from a super-sample

We can derive in our framework bounds in the same spirit of the conditional MI bound from Steinke and Zakyntinou (2020).

Let $s^* = (x_1^*, \dots, x_m^*)$ denote a $(2m)$ -sample, made of m pairs $x_i^* = (x_{i,0}, x_{i,1})$. The training sample is in the form $s = (x_1, \dots, x_m)$. The choice of s , given s^* is determined by a variable $u \in \{0, 1\}^n$, in the sense that $x_i = x_{i, u_i}^*$, where u_i determine which one of the two components of x_i^* is chosen as x_i . In practice we can write $s = s_u^*$, with $u \in \{0, 1\}^n$. We let $\bar{u} = 1 - u$ (the difference being component-wise), and $\bar{s} = s_{\bar{u}}^*$. We denote as S^* the random super-sample and we assume that each $X_i^* \in S^*$ has marginal distribution $\mathbb{P}_{X^*} = \mathbb{P}_X^{\otimes 2}$. Moreover, we let $\mathbb{P}_{\bar{U}} = \mathbb{P}_U \sim \text{Bernoulli}(\frac{1}{2})^{\otimes m}$, and we assume that $U \perp\!\!\!\perp S^*$. Note that this implies that if the super-sample is made of independent pairs ($\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$) then all the $X_i \in S$ are independent.

Proposition 50 Let $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. Assume that $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathcal{D}}(\xi)$, wrt $\mathbb{P}_{S|S^*=s^*}$, for \mathbb{P}_{S^*} -almost every s^* and $\forall w \in \mathcal{W}$. Then, we have that

$$|\mathcal{G}| \leq \xi \mathbb{E}_{\mathbb{P}_{W, S^*}} [\mathcal{D}(\mathbb{P}_{S|W, S^*}, \mathbb{P}_{S|S^*}) + \mathcal{D}(\mathbb{P}_{\bar{S}|W, S^*}, \mathbb{P}_{\bar{S}|S^*})].$$

Proof Let $\hat{g}(w, s^*, u) = \mathcal{L}_{s_u^*}(w) - \mathcal{L}_{s_{\bar{u}}^*}(w)$. Now, recalling that $S = S_U^*$ and $\bar{S} = S_{\bar{U}}^*$, we have that $\mathbb{P}_{S|S^*}$ is the law of S_U^* and $\mathbb{P}_{\bar{S}|S^*}$ is the law of $S_{\bar{U}}^*$, both under \mathbb{P}_U and given S^* . Since $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m} = \mathbb{P}_X^{\otimes 2m}$, then $S \perp\!\!\!\perp \bar{S}$. In particular $\bar{S} \perp\!\!\!\perp W$, and hence $\mathbb{P}_{\bar{S}|W} = \mathbb{P}_{\bar{S}} = \mathbb{P}_{S^*}$, so that

$$\mathbb{E}_{\mathbb{P}_{W, S^*, U}} [\mathcal{L}_{S_U^*}(W)] = \mathbb{E}_{\mathbb{P}_{W, \bar{S}}} [\mathcal{L}_{\bar{S}}(W)] = \mathbb{E}_{\mathbb{P}_{W \otimes S}} [\mathcal{L}_S(W)].$$

It follows that $\mathcal{G} = \mathbb{E}_{\mathbb{P}_{W, S^*, U}} [\hat{g}(W, S^*, U)]$. Moreover, it is shown in Rodríguez-Gálvez et al. (2021) (cf. proof of Theorem 3 therein) that

$$\begin{aligned} \mathbb{E}_{\mathbb{P}_{W, S^*, U}} [\hat{g}(W, S^*, U)] &= \mathbb{E}_{\mathbb{P}_{S^*}} [\mathbb{E}_{\mathbb{P}_{W \otimes U|S^*}} [\mathcal{L}_{S_U^*}(W)] - \mathbb{E}_{\mathbb{P}_{W, U|S^*}} [\mathcal{L}_{S_U^*}(W)]] \\ &\quad - \mathbb{E}_{\mathbb{P}_{S^*}} [\mathbb{E}_{\mathbb{P}_{W, U|S^*}} [\mathcal{L}_{S_{\bar{U}}^*}(W)] - \mathbb{E}_{\mathbb{P}_{W \otimes U|S^*}} [\mathcal{L}_{S_{\bar{U}}^*}(W)]]]. \end{aligned}$$

We hence have

$$\begin{aligned} |\mathcal{G}| &\leq \mathbb{E}_{\mathbb{P}_{S^*}} \left[\left| \mathbb{E}_{\mathbb{P}_{W \otimes U|S^*}} [\mathcal{L}_{S_U^*}(W)] - \mathbb{E}_{\mathbb{P}_{W, U|S^*}} [\mathcal{L}_{S_U^*}(W)] \right| \right. \\ &\quad \left. + \left| \mathbb{E}_{\mathbb{P}_{W \otimes U|S^*}} [\mathcal{L}_{S_{\bar{U}}^*}(W)] - \mathbb{E}_{\mathbb{P}_{W, U|S^*}} [\mathcal{L}_{S_{\bar{U}}^*}(W)] \right| \right], \end{aligned}$$

which can be rewritten as

$$|\mathcal{G}| \leq \mathbb{E}_{\mathbb{P}_{W,S^*}} [|\mathbb{E}_{\mathbb{P}_{S|S^*}}[\mathcal{L}_S(W)] - \mathbb{E}_{\mathbb{P}_{S|W,S^*}}[\mathcal{L}_S(W)]| + |\mathbb{E}_{\mathbb{P}_{\bar{S}|S^*}}[\mathcal{L}_{\bar{S}}(W)] - \mathbb{E}_{\mathbb{P}_{\bar{S}|W,S^*}}[\mathcal{L}_{\bar{S}}(W)]|]. \quad (7)$$

Now, notice that, since $\mathbb{P}_U = \mathbb{P}_{\bar{U}}$, we have $\mathbb{P}_{S|S^*} = \mathbb{P}_{\bar{S}|S^*}$. In particular, $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathcal{D}}(\xi)$ wrt $\mathbb{P}_{\bar{S}|S^*=s^*}$ as well ($\forall w \in \mathcal{W}$ and \mathbb{P}_{S^*} -a.s.). From (7) and Theorem 2, we have that

$$|\mathcal{G}| \leq \xi \mathbb{E}_{\mathbb{P}_{W,S^*}} [\mathcal{D}(\mathbb{P}_{S|W,S^*}, \mathbb{P}_{S|S^*}) + \mathcal{D}(\mathbb{P}_{\bar{S}|W,S^*}, \mathbb{P}_{\bar{S}|S^*})],$$

as requested. ■

Proposition 51 *Let $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. Assume ♣ and suppose that $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathcal{D}}(\xi)$, wrt $\mathbb{P}_{S|S^*=s^*}$, for \mathbb{P}_{S^*} -almost every s^* and $\forall w \in \mathcal{W}$. Then, we have that*

$$|\mathcal{G}| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W,S^*}} [\mathcal{D}(\mathbb{P}_{S|W_k,S^*}, \mathbb{P}_{S|S^*}) + \mathcal{D}(\mathbb{P}_{\bar{S}|W_k,S^*}, \mathbb{P}_{\bar{S}|S^*})].$$

Proof We proceed just as in the proof on Proposition 50 until the last step, where we use Theorem 4, instead of Theorem 2, to conclude. ■

We give now some explicit example of bounds that can be obtained via the above two propositions.

Definition 52 (Conditional mutual information, power information, and lautum information)

Let (Z, Z', W) be a random variable on $(\mathcal{Z} \times \mathcal{Z}' \times \mathcal{W}, \Sigma_{\mathcal{Z}} \otimes \Sigma_{\mathcal{Z}'} \otimes \Sigma_{\mathcal{W}})$. We define the conditional MI (Wyner, 1978) as

$$I(Z; Z'|W) = \mathbb{E}_{\mathbb{P}_W} [\text{KL}(\mathbb{P}_{Z,Z'|W} \| \mathbb{P}_{Z \otimes Z'|W})].$$

For $p > 1$, we define the conditional p -power information as

$$I^{(p)}(Z; Z'|W) = \mathbb{E}_{\mathbb{P}_W} [D^{(p)}(\mathbb{P}_{Z,Z'|W} \| \mathbb{P}_{Z \otimes Z'|W})].$$

Finally, we define the conditional Lautum information (Palomar and Verdú, 2008) as

$$L(Z; Z|W) = \mathbb{E}_{\mathbb{P}_W} [\text{KL}(\mathbb{P}_{Z \otimes Z|W} \| \mathbb{P}_{Z,Z|W})].$$

Proposition 53 *Let $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. On the one hand, assume that $|\ell(w, x)| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in X$. Then, we have that*

$$|\mathcal{G}| \leq 2\xi \sqrt{\frac{2I(W; S|S^*)}{m}}.$$

On the other hand, assume ♣ and suppose that $\|\nabla_w \ell(w, x)\| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in X$. Then we have

$$|\mathcal{G}| \leq 2\xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{\frac{2I(W_k; S|S^*)}{m}}.$$

Proof Assume that $|\ell| \leq \xi$. Note that $\ell(w, X)$ is ξ -SG, for all $w \in \mathcal{W}$, for $X \sim \mathbb{P}_{X|X^*=x^*}$ (for all x^*). As the elements of S are independent (even when conditioning on S^* since $U \perp\!\!\!\perp S^*$), we have that, $\forall w \in \mathcal{W}$ and $\forall s^* \in \mathcal{S}^2$, $\mathcal{L}_S(w)$ is (ξ/\sqrt{m}) -SG for $S \sim \mathbb{P}_{S|S^*=s^*}$. We can then conclude by Lemma 9 and Proposition 50, using the fact that $I(W; S|S^*) = I(W; \bar{S}|S^*)$, as \bar{s} is fully determined by s (given s^*). The proof for the chained bound is analogous. \blacksquare

The proofs for the next propositions are essentially analogous of the one of Proposition 53 and hence are omitted.

Proposition 54 Let $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$ and assume that $d_{\mathcal{X}}$ and d_S are related by (1). On the one hand, suppose that $x \mapsto \ell(w, x)$ is ξ -Lipschitz, for all $w \in \mathcal{W}$. Then, we have that

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \mathbb{E}_{\mathbb{P}_{W, S^*}} [\mathfrak{W}(\mathbb{P}_{S|S^*}, \mathbb{P}_{S|W, S^*}) + \mathfrak{W}(\mathbb{P}_{\bar{S}|S^*}, \mathbb{P}_{\bar{S}|W, S^*})].$$

On the other hand, assume \clubsuit and suppose that $x \mapsto \|\nabla_w \ell(w, x)\| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$. Then we have

$$|\mathcal{G}| \leq \frac{\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, S^*}} [\mathfrak{W}(\mathbb{P}_{S|S^*}, \mathbb{P}_{S|W_k, S^*}) + \mathfrak{W}(\mathbb{P}_{\bar{S}|S^*}, \mathbb{P}_{\bar{S}|W_k, S^*})].$$

Proposition 55 Fix $p > 1$, let $r = p/(p-1)$ and suppose that $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. On the one hand, assume that $|\ell(w, x)| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in X$. Then, we have that

$$|\mathcal{G}| \leq \frac{2e^{1/e} \sqrt{r} \xi}{\sqrt{m}} (I^{(p)}(W; S|S^*) + 1)^{1/p}.$$

On the other hand, assume \clubsuit and suppose that $\|\nabla_w \ell(w, x)\| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$. Then we have

$$|\mathcal{G}| \leq \frac{2e^{1/e} \sqrt{r} \xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} (I^{(p)}(W_k; S|S^*) + 1)^{1/p}.$$

Proposition 56 Suppose that $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. On the one hand, if $|\ell(w, x)| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in X$, then

$$|\mathcal{G}| \leq \frac{2\xi}{\sqrt{m}} \mathbb{E}_{\mathbb{P}_{W, S^*}} \left[\sqrt{\chi^2(\mathbb{P}_{S|W, S^*} \|\mathbb{P}_{S|S^*})} + \sqrt{\chi^2(\mathbb{P}_{\bar{S}|W, S^*} \|\mathbb{P}_{\bar{S}|S^*})} \right].$$

On the other hand, under the assumptions \clubsuit if $\|\nabla_w \ell(w, x)\| \leq \xi$, for all $w \in \mathcal{W}$ and all $x \in \mathcal{X}$, then

$$|\mathcal{G}| \leq \frac{2\xi}{\sqrt{m}} \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, S^*}} \left[\sqrt{\chi^2(\mathbb{P}_{S|W_k, S^*} \|\mathbb{P}_{S|S^*})} + \sqrt{\chi^2(\mathbb{P}_{\bar{S}|W_k, S^*} \|\mathbb{P}_{\bar{S}|S^*})} \right].$$

One issue with this random sub-sampling approach is that in order to control \mathcal{L}_S wrt $\mathbb{P}_{S|S^*=s^*}$, almost uniformly in s^* , one needs essentially to control the random binary variables $\ell(w, X^*)$ under $\mathbb{P}_{X|X^*=(x_0^*, x_1^*)}$ (that is $X^* = x_0^*$ with probability 1/2, and x_1^* with probability 1/2). This can be easily done in the case of the Wasserstein distance, as the Lipschitzianity guarantees \mathfrak{W} -regularity

wrt any measure. However for the subgaussianity things are more complicated, and one essentially needs to ask that ℓ is bounded.

It is however possible to restate Proposition 50 (and Proposition 51) without asking that the same regularity holds \mathbb{P}_{S^*} -a.s. The proof of both results follow closely the ones of Propositions 50 and 51, the only difference being a final application of Hölder's inequality.

Proposition 57 *Let $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. Let $p \in [1, +\infty]$ and $r = p/(p-1)$ (with the convention that $1/0 = +\infty$). Assume that $s \mapsto \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi_{s^*})$, wrt $\mathbb{P}_{S|S^*=s^*}$, for \mathbb{P}_{S^*} -almost every s^* and $\forall w \in \mathcal{W}$, where $\|\xi_{S^*}\|_{L^p(\mathbb{P}_{S^*})} = \xi$. Then, we have that*

$$|\mathcal{G}| \leq \xi \mathbb{E}_{\mathbb{P}_{W,S^*}} [|\mathfrak{D}(\mathbb{P}_{S|W,S^*}, \mathbb{P}_{S|S^*}) + \mathfrak{D}(\mathbb{P}_{\bar{S}|W,S^*}, \mathbb{P}_{\bar{S}|S^*})|^r]^{1/r}.$$

Proposition 58 *Let $\mathbb{P}_{S^*} = \mathbb{P}_{X^*}^{\otimes m}$. Let $p \in [1, +\infty]$ and $r = p/(p-1)$ (with the convention that $1/0 = +\infty$). Assume \clubsuit and suppose that $s \mapsto \nabla_w \mathcal{L}_s(w)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi_{s^*})$, wrt $\mathbb{P}_{S|S^*=s^*}$, for \mathbb{P}_{S^*} -almost every s^* and $\forall w \in \mathcal{W}$, where $\|\xi_{S^*}\|_{L^p(\mathbb{P}_{S^*})} = \xi$. Then, we have that*

$$|\mathcal{G}| \leq \xi \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W,S^*}} [|\mathfrak{D}(\mathbb{P}_{S|W_k,S^*}, \mathbb{P}_{S|S^*}) + \mathfrak{D}(\mathbb{P}_{\bar{S}|W_k,S^*}, \mathbb{P}_{\bar{S}|S^*})|^r]^{1/r}.$$

G.5. Individual-sample bounds based on random sub-sampling

We can merge together the ideas of the last two sections.

Proposition 59 *Assume that $x \mapsto \ell(w, x)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$, wrt $\mathbb{P}_{X|X^*=x^*}$, for \mathbb{P}_{X^*} -almost every x^* and $\forall w \in \mathcal{W}$. Then, we have that*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_{W,X_i^*}} [\mathfrak{D}(\mathbb{P}_{X_i|W,X_i^*}, \mathbb{P}_{X_i|X_i^*}) + \mathfrak{D}(\mathbb{P}_{\bar{X}_i|W,X_i^*}, \mathbb{P}_{\bar{X}_i|X_i^*})].$$

Proof Note that $\mathbb{P}_{X|X^*=x^*} = \mathbb{P}_{X_i|X_i^*=x^*}$. Proceeding as in the proof of Proposition 50, we can show that, for $i \in [1 : m]$,

$$\begin{aligned} & |\mathbb{E}_{\mathbb{P}_{W \otimes X_i}} [\ell(W, X_i)] - \mathbb{E}_{\mathbb{P}_{W, X_i}} [\ell(W, X_i)]| \\ & \leq \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\mathfrak{D}(\mathbb{P}_{X_i|W, X_i^*}, \mathbb{P}_{X_i|X_i^*}) + \mathfrak{D}(\mathbb{P}_{\bar{X}_i|W, X_i^*}, \mathbb{P}_{\bar{X}_i|X_i^*})]. \end{aligned}$$

We can immediately conclude by writing \mathcal{G} as in the proof of Proposition 41. \blacksquare

Proposition 60 *Assume \clubsuit and suppose that $x \mapsto \nabla_w \ell(w, x)$ has regularity $\mathcal{R}_{\mathfrak{D}}(\xi)$, wrt $\mathbb{P}_{X|X^*=x^*}$, for \mathbb{P}_{X^*} -almost every x^* and $\forall w \in \mathcal{W}$. Then, we have that*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{X_i^*, W}} [\mathfrak{D}(\mathbb{P}_{X_i|W_k, X_i^*}, \mathbb{P}_{X_i|X_i^*}) + \mathfrak{D}(\mathbb{P}_{\bar{X}_i|W_k, X_i^*}, \mathbb{P}_{\bar{X}_i|X_i^*})].$$

Proof We proceed as for proving Proposition 59, but following the proof Proposition 51 instead of 50. \blacksquare

Clearly one can generalise the two results above by using the same observations as in Propositions 57 and 58.

We can now restate all the individual-sample bounds from Section G.3 in the random sub-sampling framework.

Proposition 61 *On the one hand, if $|\ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then*

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sqrt{2I(W; X_i | X_i^*)}.$$

On the other hand, if $|\nabla_w \ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2I(W_k; X_i | X_i^*)}.$$

Proposition 62 *On the one hand, if $x \mapsto \ell(w, x)$ is ξ -Lipschitz uniformly on \mathcal{W} , then*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_{\mathcal{W}, X_i^*}} [\mathfrak{W}(\mathbb{P}_{X_i | X_i^*}, \mathbb{P}_{X_i | \mathcal{W}, X_i^*}) + \mathfrak{W}(\mathbb{P}_{\bar{X}_i | X_i^*}, \mathbb{P}_{\bar{X}_i | \mathcal{W}, X_i^*})].$$

On the other hand, assume \clubsuit . If $x \mapsto \nabla_w \ell(w, x)$ is ξ -Lipschitz uniformly on \mathcal{W} , then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{\mathcal{W}, X_i^*}} [\mathfrak{W}(\mathbb{P}_{X_i | X_i^*}, \mathbb{P}_{X_i | W_k, X_i^*}) + \mathfrak{W}(\mathbb{P}_{\bar{X}_i | X_i^*}, \mathbb{P}_{\bar{X}_i | W_k, X_i^*})].$$

Proposition 63 *Fix $p > 1$ and let $r = p/(p-1)$. On the one hand, if $|\ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then*

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m (I^{(p)}(W; X_i | X_i^*) + 1)^{1/p}.$$

On the other hand, assume \clubsuit . If $|\nabla_w \ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} (I^{(p)}(W_k; X_i | X_i^*) + 1)^{1/p}.$$

Proposition 64 *On the one hand, if $|\ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then*

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_{\mathcal{W}, X_i^*}} \left[\sqrt{\chi^2(\mathbb{P}_{X_i | \mathcal{W}, X_i^*} \| \mathbb{P}_{X_i | X_i^*})} + \sqrt{\chi^2(\mathbb{P}_{\bar{X}_i | \mathcal{W}, X_i^*} \| \mathbb{P}_{\bar{X}_i | X_i^*})} \right].$$

On the other hand, assume \clubsuit . If $|\nabla_w \ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then

$$|\mathcal{G}| \leq \frac{\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{\mathcal{W}, X_i^*}} \left[\sqrt{\chi^2(\mathbb{P}_{X_i | W_k, X_i^*} \| \mathbb{P}_{X_i | X_i^*})} + \sqrt{\chi^2(\mathbb{P}_{\bar{X}_i | W_k, X_i^*} \| \mathbb{P}_{\bar{X}_i | X_i^*})} \right].$$

Proposition 65 *On the one hand, if $|\ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then*

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \mathbb{E}_{\mathbb{P}_{\mathcal{W}, X_i^*}} \left[\text{TV}(\mathbb{P}_{X_i | X_i^*}, \mathbb{P}_{X_i | \mathcal{W}, X_i^*}) + \text{TV}(\mathbb{P}_{\bar{X}_i | X_i^*}, \mathbb{P}_{\bar{X}_i | \mathcal{W}, X_i^*}) \right].$$

On the other hand, assume \clubsuit . If $|\nabla_w \ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \left[\text{TV}(\mathbb{P}_{X_i | X_i^*}, \mathbb{P}_{X_i | W_k, X_i^*}) + \text{TV}(\mathbb{P}_{\bar{X}_i | X_i^*}, \mathbb{P}_{\bar{X}_i | W_k, X_i^*}) \right].$$

Proposition 66 *On the one hand, if $|\ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then*

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sqrt{2\mathbb{L}(W; X_i | X_i^*)}.$$

On the other hand, assume ♣. If $|\nabla_w \ell(w, x)| \leq \xi$, uniformly on \mathcal{W} and \mathcal{X} , then

$$|\mathcal{G}| \leq \frac{2\xi}{m} \sum_{i=1}^m \sum_{k=1}^{\infty} \varepsilon_{k-1} \sqrt{2\mathbb{L}(W_k; X_i | X_i^*)}.$$

G.6. Summary table

Several explicit bounds that can be derived within our general framework of Section 3 are reported in Table 1. The first column states the regularity condition required on the loss. However, we refer to the corresponding propositions for the detailed assumptions of each bound. All bounds are stated for $\xi = 1$. The last columns give the literature references for each bound, to the best of our knowledge. However, this bibliography should be taken as a mere guideline, as there might possibly be missing references. Those bounds that we could not find in the literature are marked as “New”.

Table 1: Some bounds that can be derived with the framework from Section 3

Assumption ($\forall w \in W$)	Bound	Prop	Ref
$\ell(w, X)$ 1-SG	$\sqrt{2I(W; S)/m}$	10	Russo and Zou (2019)
$\nabla_w \ell(w, X)$ 1-SG	$\sum_k \varepsilon_{k-1} \sqrt{2I(W_k; S)/m}$	13	Asadi et al. (2018)
$\ell(w, \cdot)$ 1-Lipschitz	$\mathbb{E}_{\mathbb{P}_W} [\mathbb{W}(\mathbb{P}_S, \mathbb{P}_{S W})] / \sqrt{m}$	11	Lopez and Jog (2018)
$\nabla_w \ell(w, \cdot)$ 1-Lipschitz	$\sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\mathbb{W}(\mathbb{P}_S, \mathbb{P}_{S W_k})] / \sqrt{m}$	15	New
$\ell(w, X)$ 1-SG	$e^{1/\varepsilon} \sqrt{p} (I^{(p)}(W; S) + 1)^{1/p} / \sqrt{m(p-1)}$	39	Aminian et al. (2021)
$\nabla_w \ell(w, X)$ 1-SG	$e^{1/\varepsilon} \sqrt{p} \sum_k \varepsilon_{k-1} (I^{(p)}(W_k; S) + 1)^{1/p} / \sqrt{m(p-1)}$	39	New
$\mathbb{V}_{\mathbb{P}_X}[\ell(w, X)] \leq 1$	$\mathbb{E}_{\mathbb{P}_W} [\chi^2(\mathbb{P}_{S W} \ \mathbb{P}_S)^{1/2}] / \sqrt{m}$	40	Rodríguez-Gálvez et al. (2021)
$\ \mathbb{C}_{\mathbb{P}_X}[\nabla_w \ell(w, X)]\ \leq 1$	$\sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\chi^2(\mathbb{P}_{S W_k} \ \mathbb{P}_S)^{1/2}] / \sqrt{m}$	40	New
$\ell(w, X)$ 1-SG	$\sum_i \sqrt{2I(W; X_i)/m}$	43	Bu et al. (2019)
$\nabla_w \ell(w, X)$ 1-SG	$\sum_i \sum_k \varepsilon_{k-1} \sqrt{2I(W_k; X_i)/m}$	43	Zhou et al. (2022)
$\ell(w, \cdot)$ 1-Lipschitz	$\sum_i \mathbb{E}_{\mathbb{P}_W} [\mathbb{W}(\mathbb{P}_X, \mathbb{P}_{X_i W})] / m$	44	Rodríguez-Gálvez et al. (2021)
$\nabla_w \ell(w, \cdot)$ 1-Lipschitz	$\sum_i \sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\mathbb{W}(\mathbb{P}_X, \mathbb{P}_{X_i W_k})] / m$	44	New
$\mathbb{E}_{\mathbb{P}_X} [\ell(w, X) - \bar{\ell}(w) ^{p/(p-1)}] \leq 1$	$\sum_i (I^{(p)}(W; X_i) + 1)^{1/p} / m$	45	New
$\mathbb{E}_{\mathbb{P}_X} [\ \nabla_w \ell(w, X_i) - \nabla_w \bar{\ell}(w)\ ^{p/(p-1)}] \leq 1$	$\sum_i \sum_k \varepsilon_{k-1} (I^{(p)}(W_k; X_i) + 1)^{1/p} / m$	45	New
$\mathbb{V}_{\mathbb{P}_X}[\ell(w, X)] \leq 1$	$\sum_i \mathbb{E}_{\mathbb{P}_W} [\chi^2(\mathbb{P}_{X_i W} \ \mathbb{P}_X)^{1/2}] / m$	46	New
$\ \mathbb{C}_{\mathbb{P}_X}[\nabla_w \ell(w, X)]\ \leq 1$	$\sum_i \sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\chi^2(\mathbb{P}_{X_i W_k} \ \mathbb{P}_X)^{1/2}] / m$	46	New
$ \ell \leq 1$	$\sum_i \mathbb{E}_{\mathbb{P}_W} [\text{TV}(\mathbb{P}_X, \mathbb{P}_{X_i W})] / m$	47	Rodríguez-Gálvez et al. (2021)
$\ \nabla_w \ell\ \leq 1$	$\sum_i \sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_W} [\text{TV}(\mathbb{P}_X, \mathbb{P}_{X_i W_k})] / m$	47	New
$ \ell \leq 1$	$\sum_i \sqrt{2L(W; X_i)/m}$	49	Rodríguez-Gálvez et al. (2021)
$\ \nabla_w \ell\ \leq 1$	$\sum_i \sum_k \varepsilon_{k-1} \sqrt{2L(W_k; X_i)/m}$	49	New
$ \ell \leq 1$	$2\sqrt{2I(W; S S^*)/m}$	53	Steinke and Zakyntinou (2020)
$\ \nabla_w \ell\ \leq 1$	$2\sum_k \varepsilon_{k-1} \sqrt{2I(W_k; S S^*)/m}$	53	New
$\ell(w, \cdot)$ 1-Lipschitz	$\mathbb{E}_{\mathbb{P}_{W, S^*}} [\mathbb{W}(\mathbb{P}_{S S^*}, \mathbb{P}_{S W, S^*}) + \dots] / \sqrt{m}$	54	Rodríguez-Gálvez et al. (2021)
$\nabla_w \ell(w, \cdot)$ 1-Lipschitz	$\sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, S^*}} [\mathbb{W}(\mathbb{P}_{S S^*}, \mathbb{P}_{S W_k, S^*}) + \dots] / \sqrt{m}$	54	New
$ \ell \leq 1$	$2e^{1/\varepsilon} \sqrt{p} (I^{(p)}(W; S S^*) + 1)^{1/p} / \sqrt{m(p-1)}$	55	New
$\ \nabla_w \ell\ \leq 1$	$2e^{1/\varepsilon} \sqrt{p} \sum_k \varepsilon_{k-1} (I^{(p)}(W_k; S S^*) + 1)^{1/p} / \sqrt{m(p-1)}$	55	New
$ \ell \leq 1$	$2\mathbb{E}_{\mathbb{P}_{W, S^*}} [\chi^2(\mathbb{P}_{S W, S^*} \ \mathbb{P}_{S S^*})^{1/2} + \dots] / \sqrt{m}$	56	New
$\ \nabla_w \ell\ \leq 1$	$2\sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, S^*}} [\chi^2(\mathbb{P}_{S W_k, S^*} \ \mathbb{P}_{S S^*})^{1/2} + \dots] / \sqrt{m}$	56	New
$ \ell \leq 1$	$2\sum_i \sqrt{2I(W; X_i X_i^*)/m}$	61	Haghifam et al. (2020)
$\ \nabla_w \ell\ \leq 1$	$2\sum_i \sum_k \varepsilon_{k-1} \sqrt{2I(W_k; X_i X_i^*)/m}$	61	New
$\ell(w, \cdot)$ 1-Lipschitz	$\sum_i \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\mathbb{W}(\mathbb{P}_{X_i X_i^*}, \mathbb{P}_{X_i W, X_i^*}) + \dots] / m$	62	Rodríguez-Gálvez et al. (2021)
$\nabla_w \ell(w, \cdot)$ 1-Lipschitz	$\sum_i \sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\mathbb{W}(\mathbb{P}_{X_i X_i^*}, \mathbb{P}_{X_i W_k, X_i^*}) + \dots] / m$	62	New
$ \ell \leq 1$	$2\sum_i (I^{(p)}(W; X_i X_i^*) + 1)^{1/p} / m$	63	New
$\ \nabla_w \ell\ \leq 1$	$2\sum_i \sum_k \varepsilon_{k-1} (I^{(p)}(W_k; X_i X_i^*) + 1)^{1/p} / m$	63	New
$ \ell \leq 1$	$\sum_i \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\chi^2(\mathbb{P}_{X_i W, X_i^*} \ \mathbb{P}_{X_i X_i^*})^{1/2} + \dots] / m$	64	New
$\ \nabla_w \ell\ \leq 1$	$\sum_i \sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\chi^2(\mathbb{P}_{X_i W_k, X_i^*} \ \mathbb{P}_{X_i X_i^*})^{1/2} + \dots] / m$	64	New
$ \ell \leq 1$	$2\sum_i \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\text{TV}(\mathbb{P}_{X_i X_i^*}, \mathbb{P}_{X_i W, X_i^*}) + \dots] / m$	65	Rodríguez-Gálvez et al. (2021)
$\ \nabla_w \ell\ \leq 1$	$2\sum_i \sum_k \varepsilon_{k-1} \mathbb{E}_{\mathbb{P}_{W, X_i^*}} [\text{TV}(\mathbb{P}_{X_i X_i^*}, \mathbb{P}_{X_i W_k, X_i^*}) + \dots] / m$	65	New
$ \ell \leq 1$	$2\sum_i \sqrt{2L(W; X_i X_i^*)/m}$	66	New
$\ \nabla_w \ell\ \leq 1$	$2\sum_i \sum_k \varepsilon_{k-1} \sqrt{2L(W_k; X_i X_i^*)/m}$	66	New

11. Here and in the following, “...” should be read as: “Take the same expression on the left and replace $\mathbb{P}_{S|W, S^*}$ with $\mathbb{P}_{S_i|W, S^*}$ (or $\mathbb{P}_{X_i|W, X_i^*}$ with $\mathbb{P}_{X_i|W, X_i^*}$).”

4.2 Statement of authorship

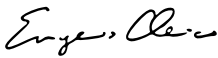
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).


Title of Paper	Chained Generalisation Bounds
Publication Status	Published
Publication Details	E. Clerico, A. Shidani, G. Deligiannidis, and A. Doucet. Chained generalisation bounds. COLT, 2022.

Student Confirmation

Student Name:	Eugenio Clerico		
Contribution to the Paper	Amitis Shidani and I equally contributed to the paper. I worked on stating and proving the main results in the paper. Amitis Shidani provided the initial idea and inspiration, helped with the writing and with some examples. George Deligiannidis and Arnaud Doucet provided helpful insights and contributed to the writing of the paper and to the checking the proofs.		
Signature		Date	24/03/2023

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:	Prof George Deligiannidis		
Supervisor comments	Eugenio's description of his contributions to the paper is fair and accurate		
Signature		Date	27/03/2023

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 5

Deterministic PAC-Bayes under gradient descent

Generalisation under gradient descent via deterministic PAC-Bayes

Eugenio Clerico*

Department of Statistics, University of Oxford

CLERICO@STATS.OX.AC.UK

Tyler Farghly*

Department of Statistics, University of Oxford

FARGHLY@STATS.OX.AC.UK

George Deligiannidis

Department of Statistics, University of Oxford

DELIGIAN@STATS.OX.AC.UK

Benjamin Guedj

Centre for Artificial Intelligence and Department of Computer Science, University College London & Inria London

B.GUEDJ@UCL.AC.UK

Arnaud Doucet

Department of Statistics, University of Oxford

DOUCET@STATS.OX.AC.UK

Abstract

We establish disintegrated PAC-Bayesian generalisation bounds for models trained with gradient descent methods or continuous gradient flows. Contrary to standard practice in the PAC-Bayesian setting, our result applies to optimisation algorithms that are deterministic, without requiring any *de-randomisation* step. Our bounds are fully computable, depending on the density of the initial distribution and the Hessian of the training objective over the trajectory. We show that our framework can be applied to a variety of iterative optimisation algorithms, including stochastic gradient descent (SGD), momentum-based schemes, and damped Hamiltonian dynamics.

1. Introduction

Effectively upper-bounding the generalisation error of modern learning algorithms is an open problem of great importance to the statistical learning theory community (Zhang et al., 2016). Originally, properties of the hypothesis space, such as VC dimension and Rademacher complexity (Vapnik, 2000; Bousquet et al., 2004; Shalev-Shwartz and Ben-David, 2014), were used to establish *worst-case* generalisation bounds, holding uniformly over all possible algorithms and training datasets. However, as these results are often vacuous in over-parameterised settings, the modern perspective focuses on algorithm and data-dependent bounds (McAllester, 1998; Bousquet and Elisseeff, 2002; Hardt et al., 2016; Xu and Raginsky, 2017; Clerico et al., 2022b; Lugosi and Neu, 2022).

Among the various approaches, the PAC-Bayesian framework (Guedj, 2019; Alquier, 2021) has obtained particularly promising empirical results (Dziugaite and Roy, 2017; Zhou et al., 2019; Pérez-Ortiz et al., 2021a,b; Biggs and Guedj, 2022b; Clerico et al., 2022a). Typically, a PAC-Bayesian bound is an upper bound on the expected population loss of a stochastic algorithm, holding with high probability on the random draw of the training dataset. Beyond the popularity of this framework is the fact that it has led to non-vacuous empirical bounds in overparameterised regimes, such as modern neural networks (Dziugaite and Roy, 2017; Zhou et al., 2019; Pérez-Ortiz et al., 2021a; Clerico et al., 2022a). Since the standard PAC-Bayesian framework relies on the randomness

* Equal contribution

of the trainable parameters, this type of analysis is typically applied to specifically designed stochastic models. For instance, in the setting of neural networks, this requires an architecture featuring stochastic weights and biases, instead of the standard deterministic ones. To extend these ideas to deterministic settings, de-randomisation techniques are used. One possibility consists in exploiting stability properties to approximate a model by randomly perturbing its parameters. While this approach has shown promising results for feed-forward neural networks (Neyshabur et al., 2018; Nagarajan and Kolter, 2019; Miyaguchi, 2019; Banerjee et al., 2020), it relies on specific architectural assumptions. Other approaches provide bounds for the predictor obtained by averaging a stochastic one, an approach started by Germain et al. (2009). However, this leads to bounds for a deterministic models with very specific structures: for instance, Letarte et al. (2019), Biggs and Guedj (2021), and Biggs and Guedj (2022a) extended this approach to get bounds for particular deterministic networks with a rather unusual erf activation function. Finally, besides PAC-Bayesian bounds in expectation, there are disintegrated results that hold with high probability on a random realisation of the stochastic model (Catoni, 2004, 2007; Blanchard and Fleuret, 2007; Alquier and Biau, 2013; Guedj and Alquier, 2013; Rivasplata et al., 2020; Viallard et al., 2021). To the best of our knowledge, this last approach has not been applied to study standard non-stochastic algorithms, such as neural networks trained via gradient-descent methods.

In the present work, we consider models trained by gradient descent-type methods and leverage the framework of disintegrated PAC-Bayesian bounds. Our starting point is noticing that often training with a deterministic optimisation scheme does still involve some randomness due to the initialisation, which features a random draw of the initial values of the parameter (for instance, this is usually the case for neural networks (Goodfellow et al., 2016)). Our analysis applies to this setting, where we show that it is possible to exploit this source of noise and obtain disintegrated PAC-Bayes bounds, holding with high probability on the random training dataset and initialisation. To the best of our knowledge, this is the first PAC-Bayesian result that directly applies to standard non-stochastic settings, without strong requirements on the model or the need for any randomness other than the initialisation. Besides, unlike bounds based on de-randomisation, ours apply with only limited assumptions made about the smoothness of the training objective and can be computed in closed form using information collected along the trajectory of the parameters during training.

We compare our bounds with other known results, including some outside the scope of the PAC-Bayesian literature. When compared to uniform stability bounds (Elisseeff, 2005; Hardt et al., 2016; Bousquet et al., 2020), we find that ours have sharper rates with respect to the size of the training dataset, and grow slower with the number of iterations. With regards to the recently popularised information-theoretic bounds (Xu and Raginsky, 2017; Negrea et al., 2019; Neu et al., 2021; Clerico et al., 2022b), ours are noticeably easier to compute and are not limited to bounds in expectation. Evaluating our bound only requires knowledge of the density of the initial distribution and of the Hessian of the training objective over the optimisation trajectory. The latter captures the flatness of the optimisation objective along the training path and can be seen to agree with the notion that flatter minima generalise better (Hochreiter and Schmidhuber, 1997; Keskar et al., 2017; Izmailov et al., 2018; He et al., 2019; Neu et al., 2021). We also highlight how this term relates to the implicit regularisation occurring in algorithms known to result in improved generalisation (Blanc et al., 2020; Damian et al., 2021). We demonstrate that this framework is easily extended to almost all iterative schemes, including stochastic variants of gradient descent (Kiefer and Wolfowitz, 1952), and iterative procedures based on auxiliary variables, like momentum schemes (Qian, 1999) or damped Hamiltonian dynamics (Hairer et al., 2006; Franca et al., 2020).

2. Notation and setting

In the standard supervised learning framework, examples are pairs $z = (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$, where x denotes the features of the example and y its label. A learning algorithm takes a training dataset $s = \{z_1, \dots, z_m\}$ of m examples and outputs a function $f : \mathcal{X} \rightarrow \mathcal{Y}$. More specifically, we consider algorithms that choose a hypothesis $h \in \mathcal{H}$, which is understood to parameterise a map $f_h : \mathcal{X} \rightarrow \mathcal{Y}$ (e.g., h could be the weights of a neural network). We will always assume that $\mathcal{H} \subseteq \mathbb{R}^d$, for some dimension $d > 0$. We call the algorithm stochastic when its output h is a random variable on \mathcal{H} whose law can depend on s .

Given a loss function $\ell : \mathcal{H} \times \mathcal{Z} \rightarrow \mathbb{R}$, we define the empirical risk on a dataset s by

$$\mathcal{L}_s(h) = \frac{1}{m} \sum_{z \in s} \ell(h, z).$$

However, what often matters is how well h predicts the labels of features outside of s . Assuming that the population of examples follows a distribution μ , the relevant quantity is the population loss,

$$\mathcal{L}_{\mathcal{Z}}(h) = \int_{\mathcal{Z}} \ell(h, z) d\mu(z).$$

Upper-bounding $\mathcal{L}_{\mathcal{Z}}$ while only having access to \mathcal{L}_s is the subject of focus in this paper and in the literature on generalisation bounds more broadly. We assume that each example in the training set s is sampled i.i.d. from μ (i.e., $s \sim \mu^m = \mu^{\otimes m}$). We are interested in upper bounds on $\mathcal{L}_{\mathcal{Z}}(h)$ (where h is the hypothesis picked by the algorithm) that hold with high probability on the random draw of s (or on (h, s) in the stochastic setting).

Our results are inspired and naturally find their place within the PAC-Bayesian framework. Although the main focus in the PAC-Bayesian literature has been on bounds in expectation, [Rivasplata et al. \(2020\)](#) and [Viallard et al. \(2021\)](#) have recently brought back interest in disintegrated bounds, which actually date back to the works of [Catoni \(2004, 2007\)](#) and [Blanchard and Fleuret \(2007\)](#). We refer to [Alquier \(2021\)](#) for an introductory exposition on PAC-Bayes that also discusses a few disintegrated results. PAC-Bayesian bounds deal with a stochastic model that, given $s \sim \mu^m$, returns a random hypothesis $h \sim \rho^s$, where the superscript s here stresses explicitly ρ 's dependence on s .¹ We will call ρ^s the *posterior* distribution and denote the joint law of (s, h) as $\mu^m * \rho^s$, that is $d(\mu^m * \rho^s)(s, h) = d\mu^m(s) d\rho^s(h)$. A disintegrated PAC-Bayesian bound is an upper bound on $\mathcal{L}_{\mathcal{Z}}(h)$ that holds with high probability over $(s, h) \sim \mu^m * \rho^s$. A fundamental ingredient in this framework is the comparison of the posterior ρ^s with a *prior* distribution π on \mathcal{H} , whose only requirement is to be data-agnostic, namely it cannot depend on the specific s used for the training. We write $\mu^m \otimes \pi$ for the law of a pair (s, h) , where $s \sim \mu^m$ and $h \sim \pi$ are independent.

The purpose of this work is to provide generalisation bounds for algorithms whose output is obtained optimising an objective $\mathcal{C}_s : \mathcal{H} \rightarrow \mathbb{R}$ via gradient-based descent methods. We can let \mathcal{C}_s depend on the training dataset s and, in practice, it can coincide with the empirical loss. However, this is not necessarily the case, as one might use a surrogate loss for the training or add some regularising term. In our analysis, we use h_t (or h_k) to denote the value of the parameters at time t (or iteration k) and similarly, we use ρ_t (or ρ_k) to denote its marginal distribution. All the measures that we consider will always be absolutely continuous with respect to the Lebesgue measure, and we will use the same notation to denote their density. The random initialisation is given by the measure ρ_0 , that we assume to have strictly positive density on the whole \mathcal{H} .

1. To be rigorous, one should actually require that $s \mapsto \rho^s$ is a Markov kernel.

3. Disintegrated PAC-Bayes for continuous-time gradient flows

We begin by considering the continuous-time dynamics of the gradient flow. While this setting is less realistic than that considered in the discrete-time analysis to follow, the analysis is considerably cleaner and will help expose some of the primary ideas of the framework we propose. We define the gradient flow $(h_0, t) \mapsto \Phi_t^s(h_0) \in \mathcal{H}$ as the solution of the dynamics

$$\partial_t \Phi_t^s(h_0) = -\nabla \mathcal{C}_s(\Phi_t(h_0)); \quad \Phi_0^s(h_0) = h_0. \quad (1)$$

We will assume that this solution exists until a fixed time horizon $T > 0$, for all initial conditions and training datasets. As h_0 and s are fixed before starting the training, we will often omit the explicit dependence on them, and simply write the solution of (1) as h_t . Given a random initialisation ρ_0 we can obtain ρ_t as the push-forward of ρ_0 under the gradient flow:

$$\rho_t = \Phi_t^{s\#} \rho_0.$$

In the following result, we take a PAC-Bayesian approach to deriving generalisation bounds by selecting ρ_0 as the prior. On the other hand, fixed $T > 0$, ρ_T depends on s through \mathcal{C}_s , and plays the role of posterior. Sampling the model's initialisation h_0 from ρ_0 and following the flow dynamics up to T , we get a hypothesis h_T that is a sample from ρ_T . Building on this last point, we can now state a PAC-Bayesian generalisation bound for an algorithm that, once drawn s and h_0 , is deterministic.

Theorem 1 *Consider the dynamics $\partial_t h_t = -\nabla \mathcal{C}_s(h_t)$, with $\mathcal{C}_s : \mathcal{H} \rightarrow \mathbb{R}$ twice differentiable. Let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable function. Fixed $\delta \in (0, 1)$ and $T > 0$, with probability at least $1 - \delta$ on the random draw $(s, h_0) \sim \mu^m \otimes \rho_0$, we have*

$$\Psi(\mathcal{L}_s(h_T), \mathcal{L}_Z(h_T)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \int_0^T \Delta \mathcal{C}_s(h_t) dt + \log \frac{\xi}{\delta}, \quad (2)$$

where Δ denotes the Laplacian with respect to h and $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h))} d\mu^m(\bar{s}) d\rho_0(h)$.

Proof The proof is based on two steps. First we keep track of how the density evolves during the training along the trajectory, by means of the ‘‘instantaneous change of variable’’ trick (Chen et al., 2018). Then we conclude with a classical Markov argument from the disintegrated PAC-Bayesian literature (Rivasplata et al., 2020), using the fact that we can explicitly express the posterior density.

For the first step, the gradient flow's continuity equation states $\partial_t \rho_t(h) = \nabla \cdot (\rho_t(h) \nabla \mathcal{C}_s(h))$, for all $h \in \mathcal{H}$, which also shows that ρ_t admits a Lebesgue density for all $t \in [0, T]$. In particular,

$$\partial_t(\rho_t(h_t)) = \partial_t \rho_t(h_t) + \nabla \rho_t(h_t) \cdot \partial_t h_t = \rho_t(h_t) \Delta \mathcal{C}_s(h_t)$$

and so

$$\log \frac{\rho_T(h_T)}{\rho_0(h_0)} = \int_0^T \Delta \mathcal{C}_s(h_t) dt.$$

For the second step, by Markov's inequality,

$$e^{\Psi(\mathcal{L}_s(h_T), \mathcal{L}_Z(h_T)) - \log \frac{\rho_T(h_T)}{\rho_0(h_T)}} \leq \frac{1}{\delta} \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h)) - \log \frac{\rho_T(h)}{\rho_0(h)}} d\mu^m(\bar{s}) d\rho_T(h).$$

with probability at least $1 - \delta$ on $(s, h_T) \sim \mu^m * \rho_T$. For all $\bar{s} \in \mathcal{Z}^m$,

$$\int_{\mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_{\mathcal{Z}}(h)) - \log \frac{\rho_T(h)}{\rho_0(h)}} d\rho_T(h) = \int_{\{\rho_T > 0\}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_{\mathcal{Z}}(h))} d\rho_0(h),$$

and so in particular

$$\Psi(\mathcal{L}_{\bar{s}}(h_T), \mathcal{L}_{\mathcal{Z}}(h_T)) \leq \log \frac{\rho_T(h_T)}{\rho_0(h_T)} + \log \frac{\xi}{\delta},$$

with probability at least $1 - \delta$ on $(s, h_T) \sim \mu^m * \rho_T$. But since sampling (s, h_T) from $\mu^m * \rho_T$ is equivalent to drawing $(s, h_0) \sim \mu^m \otimes \rho_0$ and following the dynamics up to T , we can equivalently say that the bound holds with probability at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$. Using that

$$\log \frac{\rho_T(h_T)}{\rho_0(h_T)} = \log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \log \frac{\rho_T(h_T)}{\rho_0(h_0)} = \log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \int_0^T \Delta \mathcal{C}_s(h_t) dt$$

we conclude. ■

Note that the time horizon T must be chosen a priori, as it cannot depend on the specific s and h_0 used for the training. However, adding a penalty $\log K$ to the RHS of (2) (i.e., replacing the term $\log \frac{\xi}{\delta}$ with $\log \frac{K\xi}{\delta}$) allows us to pick the best time horizon among a pool $\{T_1, \dots, T_K\}$ of K potential candidates. This follows from an elementary union argument, as for each T_k we can consider a bound holding with probability at least $1 - \delta/K$. Finally, it is worth noticing that the RHS of (2) will diverge for large T . This is due to the fact that ρ_t will tend to a sum of Dirac deltas, centred on the local minima of the objective, and hence is somehow related to the fact that standard PAC-Bayesian bounds are vacuous if the posterior is degenerate.

In order to get a more readable result from Theorem 1, one need to specialise the choice of Ψ and provide some additional hypotheses on ℓ , in a way that make possible to explicitly upperbound ξ in (14). A simple possible choice is to set $\Psi(u, v) = \sqrt{m}(v - u)$, which works for a loss function sub-Gaussian in h for each input x . If the loss ℓ is bounded in $[0, 1]$, then a tighter bound is obtained with $\Psi(u, v) = m \text{kl}(u||v)$, where $\text{kl}(u||v) = u \log \frac{u}{v} + (1 - u) \log \frac{1-u}{1-v}$ is the relative entropy between two binary Bernoulli distributions. We define $\text{kl}^{-1}(u|c) = \sup\{v \in [0, 1] : \text{kl}(u||v) \leq c\}$. We remark that these choices of Ψ are common in the PAC-Bayesian literature (Bégin et al., 2016; Alquier, 2021), and we refer to Rivasplata et al. (2020) for alternative regularity hypotheses for ℓ and choices of Ψ that still allow to control the quantity ξ appearing in our bound (14).

Corollary 2 *If $\ell(h, \cdot)$ is R -sub-Gaussian² for each $h \in \mathcal{H}$, the bound (2) takes the form*

$$\mathcal{L}_{\mathcal{Z}}(h_T) \leq \mathcal{L}_s(h_T) + \frac{1}{\sqrt{m}} \left(\log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \int_0^T \Delta \mathcal{C}_s(h_t) dt + \log \frac{1}{\delta} + \frac{R^2}{2} \right).$$

Moreover, if ℓ is bounded in $[0, 1]$ we have the tighter bound

$$\mathcal{L}_{\mathcal{Z}}(h_T) \leq \text{kl}^{-1} \left(\mathcal{L}_s(h_T) \left| \frac{\log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \log \frac{2\sqrt{m}}{\delta} + \int_0^T \Delta \mathcal{C}_s(h_t) dt}{m} \right. \right). \quad (3)$$

2. We say that $\ell(h, \cdot)$ is R -sub-Gaussian if for all $\lambda \in \mathbb{R}$ we have $\log \int_{\mathcal{Z}} e^{\ell(h, z)} d\mu(z) \leq \int_{\mathcal{Z}} \ell(h, z) d\mu(z) + \frac{R^2 \lambda^2}{2}$.

The presence of kl^{-1} in (3) might puzzle some readers. It was introduced by Langford and Seeger (2001) and Maurer (2004) to the PAC-Bayesian community, and has by now become standard. A useful property is that $\text{kl}^{-1}(u|c) \leq \min\{2(u+c), u + \sqrt{c/2}\}$ (Lemma 7 in Appendix B). This translates to the fact that Corollary 2 leads to a fast-rate bound $O(\log m/m)$ for small enough values of the empirical loss, $\mathcal{L}_s \leq O(1/m)$; see also Tolstikhin and Seldin (2013) and the discussion and references therein.

As a final comment for this section, an interesting feature of the bound (2) is the integral of the Laplacian of the optimisation objective along the training path. Under the gradient flow dynamics, $\Delta\mathcal{C}_s(h_t)$ is keeping track of how the probability density is locally varying. Indeed, from a PAC-Bayesian perspective, for the bound to be small we need to end up in some point where the posterior density is not too high compared to the initial one. If we follow a trajectory characterised by a large Laplacian, we see a sharp increase of the density. Intuitively, we can picture the situation as if we were attracting the nearby paths and bringing further *probability mass* around us. In such a case, the final ρ_T is likely to be much larger than the initial ρ_0 and lead to a loose bound. Note that we can rewrite

$$\int_0^T \Delta\mathcal{C}_s(h_t) dt = \log \frac{\|\nabla\mathcal{C}_s(h_0)\|}{\|\nabla\mathcal{C}_s(h_T)\|} - \int_{h_{[0:T]}} \nabla \cdot \tau(h) \|\delta h\|. \quad (4)$$

Here $\tau(h) = -\frac{\nabla\mathcal{C}_s(h)}{\|\nabla\mathcal{C}_s(h)\|}$ is the unit tangent vector to the trajectory in h , and $\int_{h_{[0:T]}} \dots \|\delta h\|$ denotes the line integral along the path $h_{[0:T]}$. The last term has a clear meaning, as it quantifies how much $h_{[0:T]}$ is attracting the nearby trajectories. We refer to Appendix C for a derivation of (4).

4. Discrete time dynamics

When trying to restate the results of the previous section for a discrete time gradient descent algorithm, the main obstacle comes from the fact that we cannot anymore use the gradient flow continuity equation to keep track of the density change along the trajectory. However, the change in density can still be computed exactly as long as we can ensure that the update map is injective and differentiable. To make things more concrete, let G_η denote the map

$$G_\eta(h) = h - \eta\nabla\mathcal{C}_s(h).$$

We fix the total number of steps $K \geq 1$ and a training schedule $\{\eta_k\}_{k=0}^{K-1}$. We consider the updates

$$h_{k+1} = G_{\eta_k}(h_k).$$

As usual we denote as ρ_k the law of h_k , and we assume that ρ_0 admits a positive Lebesgue density on the whole \mathcal{H} .

Theorem 3 *Consider the dynamics $h_{k+1} = G_{\eta_k}(h_k)$. For each dataset $s \in \mathcal{Z}^m$, denote as A_s a Borel set on which \mathcal{C}_s is twice-differentiable and M -smooth³, with $\sup_k \eta_k \leq 1/(2M)$. Let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable function. Fix $K \in \mathbb{N}$ and choose $\delta \in (0, 1)$, such that the trajectory*

3. A function f is said M -smooth on a set A if its Jacobian is M -Lipschitz, that is for any $h, h' \in A$ we have $\|\nabla f(h) - \nabla f(h')\| \leq M\|h - h'\|$.

$\{h_k\}_{k=0}^{K-1}$ lies in A_s with probability at least $1 - \delta/2$, under $(s, h_0) \sim \mu^m \otimes \rho_0$. Then, with a probability of at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$,

$$\Psi(\mathcal{L}_s(h_T), \mathcal{L}_Z(h_T)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_k)} - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) + \log \frac{2\xi}{\delta} + \delta/2, \quad (5)$$

where $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h))} d\mu^m(\bar{s}) d\rho_0(h)$.

We refer to Appendix A for the proof. We note that the term $-\text{tr} \log(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k))$ in (5) can be upper bounded in various ways, using the fact that \mathcal{C}_s is smooth around h_k . From this, we see that the continuous time bound of Theorem 1 is recovered when one let the learning rates tend to 0.

Lemma 4 *With the notation of Theorem 3, let $h_k \in A_s$. Then*

$$-\text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) \leq \eta_k \Delta \mathcal{C}_s(h_k) + \eta_k^2 \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{F}}^2 \leq \frac{3}{2} \eta_k \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{TR}},$$

where $\|\cdot\|_{\text{F}}$ is the Frobenius norm and $\|\cdot\|_{\text{TR}}$ the trace norm.⁴

From the final inequality, it follows that this term scales at worst as $O(d)$, as the smoothness assumption ensures that $\eta_k \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{TR}} \leq d/2$. However, it is likely that in many cases this translates in an over-pessimistic estimate. For instance, we will show in the next section that for a simple random feature model one can upper-bound $\|\nabla^2 \mathcal{C}_s\|_{\text{TR}}$ with a term that is of order $O(1)$ for large d . It is also worth mentioning that $-\text{tr} \log(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k))$ can be directly expressed in terms of the spectrum $\{\lambda_i\}_{i=1}^d$ of $\nabla^2 \mathcal{C}_s(h_k)$, as we have

$$-\text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) = \sum_{i=1}^d \log \frac{1}{1 - \eta_k \lambda_i}.$$

5. Discussion and examples

In this section, we discuss various aspects and consequences of the results given in the previous sections. As part of this, we consider two examples that allow for further theoretical inspection.

5.1. Random feature model

To start, we investigate how $\|\nabla^2 \mathcal{C}_s\|_{\text{TR}}$ scales with d in practice, by considering a simple feature model for classification. In this setting, we let $\mathcal{X} \subseteq \mathbb{R}^p$ and $\mathcal{Y} = \{1, \dots, q\}$. We consider the class of mappings $F_h : \mathcal{X} \rightarrow \mathbb{R}^q$ defined as

$$F_h(x) = \frac{1}{\sqrt{d}} h \Phi(x), \quad (5)$$

for $h \in \mathbb{R}^{q \times d}$, where $\Phi : \mathbb{R}^p \rightarrow \mathbb{R}^d$ is a feature map fixed during training. The label predicted by the model will be $f_h = \text{argmax}_i F_h^i$, the index of the largest component of F_h . We consider

4. For a matrix U with singular values $\{\sigma_i\}$, let $\|U\|_{\text{TR}} = \sum_i \sigma_i$ and $\|U\|_{\text{F}} = \sqrt{\sum_i \sigma_i^2} = \text{tr}[UU^\top]^{1/2}$.

5. The factor $1/\sqrt{d}$ is a standard choice for this kind of models, as it brings $\|h/\sqrt{d}\| \sim O(1)$ at initialisation, when the components of h are independently initialised $\mathcal{N}(0, 1)$ (e.g., see Ghorbani et al., 2021 for random feature models).

an optimisation objective $\mathcal{C}_s(h) = \frac{1}{m} \sum_{z \in s} \hat{\ell}(F_h(x), y)$, where $\hat{\ell}$ denotes the cross-entropy loss $\hat{\ell}(F, y) = \log(\sum_i e^{F^i}) - F^y$, with F^i and F^y respectively denoting the i -th and y -th component of F . Since this is convex in F and the model is linear in h , we get that $\hat{\ell}(F_h(x), y)$ is itself convex in h , and so $\|\nabla^2 \mathcal{C}_s\|_{\text{TR}} = \Delta \mathcal{C}_s$. The Laplacian can be explicitly evaluated (see Appendix D.1):

$$\Delta \mathcal{C}_s(h) = \frac{1}{m} \sum_{z \in s} \frac{1}{d} \|\Phi(x)\|^2 \left(1 - \frac{\sum_i e^{2F^i(x)}}{(\sum_i e^{F^i(x)})^2} \right) \leq \frac{1}{m} \sum_{z \in s} \frac{1}{d} \|\Phi(x)\|^2.$$

To further investigate this quantity, we consider the setting of random feature models (Rahimi and Recht, 2007; Mei and Montanari, 2022), where the features are given by $\Phi(x) = \phi(Wx)$. Here ϕ is a non-linearity acting component-wise, and W is a $d \times q$ matrix whose rows are sampled independently. Assuming that $|\phi(r)| \leq |r|$, we can further bound the Laplacian by $\frac{1}{dm} \|W\|^2 \sum_{z \in s} \|x\|^2$, with $\|W\|$ denoting the matrix operator norm of W . If we consider the case where the components of W are independently drawn from a standard Gaussian distribution, we get that $\|W\| = O(\sqrt{d})$ for large d , and so $\Delta \mathcal{C}_s(h) = O(1)$. Thus, it is concluded that the trajectory-dependent term in (5) can be controlled using

$$\text{tr} \log(\text{Id} - \eta \nabla^2 \mathcal{C}_s(h)) = O(\eta).$$

5.2. Wide neural networks

Next, we investigate the first term of the bound, $\log \rho_0(h_0)/\rho_0(h_k)$, by considering the setting of wide feed-forward neural networks. For simplicity, we assume that each layer has the same activation and the hidden layers have the same width n , and that each of the weights and bias parameters are initialised with a centred Gaussian distribution with variance σ_w^2/n and σ_b^2 , respectively. We also assume the learning rate decays faster than $O(1/n)$ so that the NN enters the NTK regime in the large width limit (see Appendix D.2). We consider the case of an optimisation objective quadratic in the network's output.

Borrowing the analysis of Lee et al. (2020), we obtain two properties of the large n setting: (i) with high probability, h_0 (the initial value of all weights and biases) is such that \mathcal{C}_s is smooth and bounded in a region around it and (ii) with the same probability, gradient descent stays close to initialisation. In combination with Theorem 3, we obtain the generalisation bound in the proposition that follows. We refer to Appendix D.2 for the proof.

Proposition 5 (Informal statement) *For each n , consider a training schedule $\{\eta_k\}_{k=0}^{K-1}$ such that $\eta_k \leq \eta_{\max}/n$, where $\eta_{\max} > 0$ is a constant independent of n . Under suitable regularity conditions, for any $\delta \in (0, 1)$, there exists $n_{\min}^\delta \in \mathbb{N}$ such that whenever $n \geq n_{\min}^\delta$, the assumptions of Theorem 3 are satisfied. In particular, with a probability of at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$,*

$$\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) \leq \frac{C}{\sigma_w^2} (C/2 + \|h_0\| n^{1/2}) - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) + \log \frac{2\xi}{\delta} + \delta/2,$$

where $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_s(h), \mathcal{L}_Z(h))} d\mu^m(\bar{s}) d\rho_0(h)$ and $C > 0$ is some constant that is independent of n , δ and m .

Note that $\|h_0\| \sim (nL)^{1/2}$ and thus, if we were to employ the bound (3) of Corollary 2, the first term of the above bound would contribute a term of order $O(nL^{1/2}/m)$. This is notably sharper

than what follows from the naive deduction that it scales at worst linearly with the number of parameters, which is $d = O(n^2L)$ in this case. As for the Hessian-dependent term, although in general it is difficult to characterise the spectrum of the Hessian, empirical studies on deep neural networks suggest that in the later stages of training, the Hessian has only a few dominant positive Hessian eigenvalues (Sagun et al., 2018; Ghorbani et al., 2019; Sankar et al., 2020), meaning that $\|\nabla^2\mathcal{C}_s(h_k)\|_{\text{TR}}$ will typically be much smaller than d times the largest eigenvalue of the Hessian.

5.3. Implicit regularisation in first-order methods

In this section, we consider some modifications of gradient descent that are known to lead to better generalisation and we relate this improvement to the bound given in Theorem 3.

The role of noise in both optimisation and generalisation is a topic that has received considerable attention recently. The Langevin dynamics, as well as its discrete-time approximations, have proved to be a popular model for investigating this (Raginsky et al., 2017; Mou et al., 2018; Farghly and Rebeschini, 2021; Erdogdu et al., 2018). The evolution of the Langevin diffusion is most often described by a stochastic differential equation of the form,

$$dh_t = -\nabla\mathcal{C}_s(h_t)dt + \sigma dB_t, \tag{6}$$

where B_t is a Brownian motion process. Its generalisation properties have been studied using uniform stability and information-theoretic bounds (see sections 7.2 and 7.3). By analysing the associated Fokker-Planck equation (Pavliotis, 2014), one can show that the evolution of the marginal density is identical to that of the deterministic flow,

$$\partial_t \hat{h}_t = -\nabla \left(\mathcal{C}_s(\hat{h}_t) + \frac{\sigma^2}{2} \log \hat{\rho}_t(\hat{h}_t) \right), \quad \hat{h}_0 \sim \rho_0, \tag{7}$$

where we denote as $\hat{\rho}_t$ the density of \hat{h}_t . This idea was recently exploited in Song et al. (2021) in the context of generative modelling, for example. We compare this to Theorem 1 which, with the choice of $\Psi(u, v) = \frac{2}{\sigma^2}(v - u)$ (for an arbitrary $\sigma > 0$) and with a bounded loss $\ell \subseteq [0, 1]$, leads to a bound in the form

$$\mathcal{L}_{\mathcal{Z}}(h_T) \leq \mathcal{L}_s(h_T) + \frac{1}{4m\sigma^2} + \frac{\sigma^2}{2} \left(\log \frac{\rho_T(h_T)}{\rho_0(h_T)} + \log \frac{1}{\delta} \right).$$

Using this bound as an optimisation objective is equivalent to following the dynamics (7) with $\mathcal{C}_s(h) = \mathcal{L}_s(h) - \frac{\sigma^2}{2} \log \rho_0(h)$.

On another note, it has been shown that applying uniformly distributed label noise during training can improve generalisation results (Blanc et al., 2020; Damian et al., 2021). Damian et al. (2021) established that in certain settings this improvement is characterised by the following implicit regularisation term that is added to the optimisation objective:

$$-\frac{\sigma^2}{2B} \text{tr} \log \left(1 - \frac{\eta}{2} \nabla^2 \mathcal{C}_s \right).$$

Here σ is the scale of the label noise and B is the batch size. Indeed, up to a scaling factor, this is precisely the term that is summed over in the bound given in Theorem 3.

However, we note that these algorithms are not optimising our PAC-Bayesian bounds. Indeed, if one were to try to use the bound as an optimisation objective, then the generalisation guarantee provided by our theory would change (as it would involve the Hessian of this new objective). We defer to future work for deeper analysis of how our bounds relate to the generalisation properties of these algorithm.

6. Extension to other algorithms

While we have focused on gradient descent, the analysis can be extended to any iterative scheme of the form

$$h_{k+1} = h_k + V_s(h_k; k),$$

where $V_s : \mathcal{H} \times \mathbb{N} \rightarrow \mathcal{H}$ can be any iteration-dependent vector field. This leads to the following generalisation of Theorem 3, which differs only the trajectory dependent sum, where the Jacobian of the vector field now replaces the Hessian of \mathcal{C}_s . As one might expect, a similar result is found for continuous flows (see Theorem 10 in Appendix E).

Theorem 6 *Consider the dynamics $h_{k+1} = h_k + V_s(h_k; k)$. For each dataset $s \in \mathcal{Z}^m$, denote as A_s a Borel set on which V_s is differentiable and M -Lipschitz, with $M \leq 1/2$. Let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable function. Fix $K \in \mathbb{N}$ and choose $\delta \in (0, 1)$, such that the trajectory $\{h_k\}_{k=0}^{K-1}$ lies in A_s with probability at least $1 - \delta/2$, under $(s, h_0) \sim \mu^m \otimes \rho_0$. Then, with a probability of at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$,*

$$\Psi(\mathcal{L}_s(h_K), \mathcal{L}_{\mathcal{Z}}(h_K)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_K)} - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} + \nabla V_s(h_k; k) \right) + \log \frac{2\xi}{\delta} + \delta/2,$$

where $\nabla V_s(h; k)$ denotes the Jacobian of $V_s(\cdot; k)$ and $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_{\mathcal{Z}}(h))} d\mu^m(\bar{s}) d\rho_0(h)$.

Stochastic gradient descent An immediate corollary of this is that our theory applies to noisy variants of gradient descent with little modification. For example, we can consider a version of gradient descent that only evaluates \mathcal{C} on a mini-batch $s_k \subset s$ at each iteration $k \in \mathbb{N}$, by simply setting $V_s(h; k) = -\eta_k \nabla \mathcal{C}_{s_k}(h)$. The resulting generalisation bound applies identically for stochastic variants of this scheme, including stochastic gradient descent, where the resulting bound is a function of the instance of the sampled mini-batches. To make things more explicit, we consider a surrogate loss function $\hat{\ell} : \mathcal{Z} \times \mathcal{H} \rightarrow \mathbb{R}$ and for a batch $s_k \subset s$, we write $\mathcal{C}_{s_k}(h) = \frac{1}{|s_k|} \sum_{z \in s_k} \hat{\ell}(z, h)$. For a sequence of batches $\{s_k\}$ (potentially randomly selected), we consider the dynamics $h_{k+1} = h_k - \eta_k \nabla \mathcal{C}_{s_k}(h_k)$. Then, under suitable smoothness assumptions for $\hat{\ell}$, we can derive from Theorem 6 the following bound

$$\Psi(\mathcal{L}_s(h_K), \mathcal{L}_{\mathcal{Z}}(h_K)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_K)} - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} + \eta_k \Delta \mathcal{C}_{s_k}(h_k) \right) + \log \frac{2\xi}{\delta} + \frac{\delta}{2},$$

which holds with a probability higher than $1 - \delta$ on the randomness of the training dataset s , the initialisation h_0 , and the choice of the batches. We refer to Proposition 11 in Appendix E.1 for a rigorous statement.

Momentum dynamics We can also use Theorem 6 to consider settings in which auxiliary variables are used to compute the update. We do this by replacing h_k with the pair (h_k, v_k) , where v_k denotes the auxiliary variable. Indeed, this setting applies to a wide range of optimisation schemes. Note that in this scenario, the initial density ρ_0 must refer to the pair (h_0, v_0) .

An interesting example consists in the momentum dynamics

$$h_{k+1} = h_k + v_{k+1}, \quad v_{k+1} = \mu_k v_k - \eta_k \nabla \mathcal{C}_s(h_k),$$

for some momentum schedule $\{\mu_k\}$. In such a case we obtain a high probability bound in the form

$$\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) \leq \log \frac{\rho_0(h_0, v_0)}{\rho_0(h_K, v_K)} - d \sum_{k=0}^{K-1} \log \frac{1}{\mu_k} + \log \frac{2\xi}{\delta} + \frac{\delta}{2}.$$

We refer to Appendix E.2 for further details and discussion.

Damped Hamiltonian dynamics For damped Hamiltonian dynamics (França et al., 2020), we can exploit the fact that the joint density of the pair ‘position-momentum’ pair (h, v) is conserved under the Hamiltonian flow, a property that is preserved for discrete time-steps by suitable symplectic integrators (Hairer et al., 2006). Using this property, we obtain bounds for discrete-time algorithms without any smoothness assumptions on the objective \mathcal{C}_s , other than twice differentiability. We refer to Appendix F for the details.

7. Comparison with the literature

7.1. Comparison with other PAC-Bayesian bounds

Contrary to many PAC-Bayesian results, our bound has the remarkable feature of applying to neural networks with deterministic parameters, trained via standard gradient-descent methods. Yet, this is not a complete novelty. For instance, a few previous works in the literature propose to study the generalisation of a deterministic neural network by analysing, via PAC-Bayesian methods, a noisy stochastic perturbation of it. This idea was exploited by Neyshabur et al. (2018), and later Biggs and Guedj (2022a), for a L -layer fully connected neural network with a 1-Lipschitz homogeneous activation function, and a 1-Lipschitz loss. In these works, margin arguments were combined with PAC-Bayesian techniques to find a bound that (up to logarithmic factors) scales as

$$\mathcal{L}_Z(h) - \mathcal{L}_s^\gamma(h) \leq O\left(\sqrt{\frac{dn\Gamma}{\gamma^2 m}}\right), \tag{8}$$

where n is the width of the network, \mathcal{L}_s^γ the margin empirical loss with margin $\gamma > 0$,⁶ and $\Gamma = \sum_{l=1}^L \left(\|W_l\|_F^2 \prod_{l' \neq l} \|W_{l'}\|\right)$, with $\{W_l\}_{l=1}^L$ the weights of the network, $\|\cdot\|$ denoting the spectral norm, and $\|\cdot\|_F$ the Frobenius norm. One of the main issues of this result is R ’s exponential dependence on the depth, due to the product of the norm of the weights. On the other hand, our bounds involve the Hessian term, which are at most of order d (see Lemma 4), and the contribution $\log \frac{\rho_0(h_0)}{\rho_0(h_T)}$. When the weights are independently initialised as $\mathcal{N}(0, 1)$, this last term is

6. The margin γ is a standard measure of the confidence of the network’s prediction. We refer to Neyshabur et al. (2018) or Biggs and Guedj (2021) for a definition of the margin loss. Note however that we always have $\mathcal{L}_s^\gamma \geq \mathcal{L}_s$.

upperbounded by $\frac{1}{2} \sum_{l=1}^L \|W_l\|_F^2$. Moreover, as discussed in Section 4, for low values of \mathcal{L}_s , our bound can have a fast-rate dependence of $O(1/m)$ with the training dataset size.

Later, building on the ideas from Neyshabur et al. (2018), Nagarajan and Kolter (2019) obtained a bound that does not suffer of the exponential dependence on the depth. However, this comes at the price of a bound that scales inversely with the smallest absolute value of the pre-activations on the training data, a fact that leads to vacuous bounds in practice. We mention that a result similar to that of Neyshabur et al. (2018) had previously been established by Bartlett et al. (2017), without the use of PAC-Bayesian techniques.

The bounds discussed so far only take into account the final output of the algorithm, while our result looks at the evolution of the model during the training. A similar spirit is shared by Miyaguchi (2019), where the author focuses on the continuous time setting and studies the evolution of the generalisation gap under the gradient flow training dynamics. Applying this result to a multilayer network, it is possible to re-derive (8) under slightly weaker assumptions.

Other PAC-Bayesian results that hold for deterministic models usually cannot be applied to standard training algorithms, as they require strong, and often unusual, assumptions on the model architecture (e.g., Letarte et al., 2019; Germain et al., 2009; Biggs and Guedj, 2021), or the sample of the parameters from the posterior distribution (e.g., Zantedeschi et al., 2021; Viallard et al., 2021; Rivasplata et al., 2020).

7.2. Comparison with the stability literature

Another method for obtaining algorithm-dependent generalisation bounds is the framework of uniform stability Hardt et al. (2016); Pensia et al. (2018); Farghly and Rebeschini (2021); Raj et al. (2023), proposed by Elisseeff (2005). This approach has recently received attention due to its application to fundamental optimisation methods, such as gradient descent and its stochastic counterpart (Hardt et al., 2016), while other works have leveraged it to obtain bounds in high-probability (Feldman and Vondrak, 2018, 2019; Bousquet et al., 2020). Hardt et al. (2016) considered the training of SGD on non-convex training objectives. Under the assumption that \mathcal{C}_s is M -smooth in h (uniformly on \mathcal{Z}), ℓ is L -Lipschitz and bounded in $[0, 1]$, and the step-size satisfies $\eta_k \leq c/(k+1)$, the combined analysis of Hardt et al. (2016) and Bousquet et al. (2020) leads to the bound,

$$\mathcal{L}_{\mathcal{Z}}(h) \lesssim \mathcal{L}_s(h) + \frac{(K/c)^{\frac{Mc}{Mc+1}} \log(m/\delta)}{Mm} + \sqrt{\frac{\log \delta^{-1}}{m}}, \quad (9)$$

which holds with probability $\delta \in (0, 1)$.⁷

To compare with these results, we recall that under the same assumptions, Theorem 3 leads to the bound,

$$\mathcal{L}_{\mathcal{Z}}(h_K) \lesssim \mathcal{L}_s(h_K) + \frac{\log \frac{\rho_0(h_0)}{\rho_0(h_K)} + \sum_{k=0}^{K-1} (\eta_k \Delta \mathcal{C}_s(h_k) + \eta_k^2 \|\nabla^2 \mathcal{C}_s(h_k)\|_F^2) + \log(2m/\delta)}{m},$$

for GD and, using a Proposition 11, a similar bound is obtained for SGD.

As a first point of comparison, we note that our analysis does not require the Lipschitz assumption and only requires smoothness to hold along the path of GD with high probability. Additionally, our analysis holds in both the stochastic and non-stochastic setting, whereas the technique used by

7. The notation \lesssim denotes that the inequality holds up to a multiplicative constant.

Hardt et al. (2016) fundamentally requires random mini-batches to have bounds that decay with m . While the bound of (9) decays at a rate of $m^{-1/2}$, our bound decays faster with rates $\log(m)/m$ (given that the training loss is small – see the discussion following Corollary 2). The fact that $\sum_{k=0}^{K-1} \eta_k \sim c \log K$ suggests that our bound may scale better with K , though this would require the $\Delta \mathcal{C}_s(h_k)$ and $\|\nabla^2 \mathcal{C}_s(h_k)\|_F$ terms to not grow too quickly with k . In the worst case, the smoothness can be used to upper bound these terms by $3dM/2$.

Lastly, we note that one of the main criticisms to the uniform stability approach is that it is solely related to the algorithm and does not consider specifics of the data or the distribution of the labels, raising doubts on its ability to distinguish whether a model has been trained on true or random labels (Zhang et al., 2016). On the other hand, our bound can depend on the data distribution through the optimisation objective \mathcal{C}_s and its landscape along the training trajectory.

7.3. Comparison with information-theoretic bounds

Another popular direction within the literature on generalisation bounds uses ideas from information theory to upper bound the expected generalisation error in terms of the mutual information (Xu and Raginsky, 2017; Russo and Zou, 2019). This has been particularly practical for developing data-dependent bounds for noisy iterative methods, such as stochastic gradient Langevin dynamics and SGD (Mou et al., 2018; Negrea et al., 2019; Neu et al., 2021).

The general approach to this requires controlling the mutual information between the training data and the update of each iterate. Therefore, this technique is restricted to settings where noise is applied at each iteration, and the bounds explode when the amount of noise is reduced. To apply this to GD and SGD, Neu et al. (2021) consider a surrogate model trained by a Gaussian perturbation of these iterates. They show that when the loss is R -sub-Gaussian, the expected generalisation error is bounded by

$$\mathbb{E} \mathcal{L}_{\mathcal{Z}}(h_K) \lesssim \mathbb{E} \mathcal{L}_s(h_K) + \sqrt{\frac{R^2}{m} \sum_{k=1}^k \eta_k^2 \mathbb{E} A(h_k) + |\mathbb{E} B(h_K)|},$$

where $A(h)$ and $B(h)$ measures the sensitivity of the gradient and the loss function, respectively, to perturbations in the parameters and dataset at h . A notable difference between this technique and our method is that this can only provide bounds in expectation. This comes with the downside that the right-hand side can usually not be computed exactly and the expectation of A and B should be approximated using a Monte Carlo average. In contrast, our bound is based on the instance of the optimisation trajectory, it can be computed exactly. Furthermore, our bounds have better dependence on m but worse dependence on η_k .

8. Conclusion

We derive novel high-probability generalisation bounds for models learned via optimisation algorithms such as gradient descent. Contrary to the standard PAC-Bayesian framework, our guarantees apply to models whose only randomness lies in the initialisation without requiring any *de-randomisation* step. To the best of our knowledge, our results are the first to leverage the disintegrated PAC-Bayesian framework to analyse such settings. We make this explicit by stating a bound that holds for wide neural networks trained via gradient descent.

A strength of our bounds is that it assumes little about the model or training procedure. For the continuous gradient flow dynamics, we require only that the optimisation objective be twice differentiable. For the discrete-time algorithm, we require smoothness and twice differentiability only in high probability on the trajectory. Additionally, we show that our results can be extended to settings more general than gradient descent and give explicit bounds for SGD, momentum schemes, and damped Hamiltonian dynamics. We foresee that this should motivate further work into developing generalisation bounds for other optimisation algorithms.

A promising direction for future work could be designing computationally efficient methods for computing these bounds. We would also like to evaluate the tightness of our guarantees and compare them with other results known in the literature with thorough empirical investigation. We also believe that our results can be improved by identifying more easily verifiable assumptions to make the framework more broadly applicable.

Acknowledgments

Eugenio Clerico is partially supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the grant EP/R513295/1 (DTP scheme) and Arnaud Doucet by EPSRC CoSInES EP/R034710/1. Tyler Farghly was supported by EPSRC EP/T5178 and by the DeepMind scholarship. Benjamin Guedj and Arnaud Doucet acknowledge support of the UK Defence Science and Technology Laboratory (DSTL) and EPSRC grant EP/R013616/1. This is part of the collaboration between US DOD, UK MOD and UK EPSRC under the Multidisciplinary University Research Initiative. Benjamin Guedj acknowledges partial support from the French National Agency for Research, grants ANR-18-CE40-0016-01 and ANR-18-CE23-0015-02. The authors would like to thank Jake Fawkes, Shahine Bouabid, Umut Şimşekli, and Patrick Rebeschini for the valuable comments and suggestions.

References

- P. Alquier. User-friendly introduction to PAC-Bayes bounds. *arXiv:2110.11216*, 2021.
- P. Alquier and G. Biau. Sparse single-index model. *Journal of Machine Learning Research*, 14(1), 2013.
- A. Banerjee, T. Chen, and Y. Zhou. De-randomized PAC-Bayes margin bounds: Applications to non-convex and non-smooth predictors. *arXiv:2002.09956*, 2020.
- P.L. Bartlett, D.J. Foster, and M.J. Telgarsky. Spectrally-normalized margin bounds for neural networks. *NeurIPS*, 2017.
- F. Biggs and B. Guedj. Differentiable PAC-Bayes objectives with partially aggregated neural networks. *Entropy*, 23(10), 2021.
- F. Biggs and B. Guedj. On margins and derandomisation in PAC-Bayes. *AISTATS*, 2022a.
- F. Biggs and B. Guedj. Non-vacuous generalisation bounds for shallow neural networks. *ICML*, 2022b.
- G. Blanc, N. Gupta, G. Valiant, and P. Valiant. Implicit regularization for deep neural networks driven by an Ornstein-Uhlenbeck like process. *COLT*, 2020.
- G. Blanchard and F. Fleuret. Occam’s hammer. *COLT*, 2007.
- O. Bousquet and A. Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2, 2002.
- O. Bousquet, S. Boucheron, and G. Lugosi. *Introduction to Statistical Learning Theory*. Springer, 2004.
- O. Bousquet, Y. Klochkov, and N. Zhivotovskiy. Sharper bounds for uniformly stable algorithms. *COLT*, 2020.
- L. Bégin, P. Germain, F. Laviolette, and J.F. Roy. PAC-Bayesian bounds based on the Rényi divergence. *AISTATS*, 2016.
- O. Catoni. *Statistical learning theory and stochastic optimization. Ecole d’été de probabilités de Saint-Flour XXXI-2001*. Springer, 2004.
- O. Catoni. PAC-Bayesian supervised classification: The thermodynamics of statistical learning. *IMS Lecture Notes Monograph Series*, 2007.
- R.T.Q. Chen, Y. Rubanova, J. Bettencourt, and D.K. Duvenaud. Neural ordinary differential equations. *NeurIPS*, 2018.
- E. Clerico, G. Deligiannidis, and A. Doucet. Conditionally Gaussian PAC-Bayes. *AISTATS*, 2022a.
- E. Clerico, A. Shidani, G. Deligiannidis, and A. Doucet. Chained generalisation bounds. *COLT*, 2022b.

- A. Damian, T. Ma, and J.D. Lee. Label noise SGD provably prefers flat global minimizers. *NeurIPS*, 2021.
- G.K. Dziugaite and D.M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *UAI*, 2017.
- A. Elisseeff. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6, 2005.
- M. Erdogdu, L. Mackey, and O. Shamir. Global non-convex optimization with discretized diffusions. *NeurIPS*, 2018.
- T. Farghly and P. Rebeschini. Time-independent generalization bounds for SGLD in non-convex settings. *NeurIPS*, 2021.
- V. Feldman and J. Vondrak. Generalization bounds for uniformly stable algorithms. *NeurIPS*, 2018.
- V. Feldman and J. Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. *COLT*, 2019.
- G. França, J. Sulam, D. Robinson, and R. Vidal. Conformal symplectic and relativistic optimization. *NeurIPS*, 2020.
- P. Germain, A. Lacasse, F. Laviolette, and M. Marchand. PAC-Bayesian learning of linear classifiers. *ICML*, 2009.
- B. Ghorbani, S. Krishnan, and Y. Xiao. An investigation into neural net optimization via Hessian eigenvalue density. *ICML*, 2019.
- B. Ghorbani, S. Mei, T. Misiakiewicz, and A. Montanari. Linearized two-layers neural networks in high dimension. *The Annals of Statistics*, 49, 2021.
- I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.
- B. Guedj. A primer on PAC-Bayesian learning. *Proceedings of the Second Congress of the French Mathematical Society*, 2019.
- B. Guedj and P. Alquier. PAC-Bayesian estimation and prediction in sparse additive models. *Electronic Journal of Statistics*, 7, 2013.
- E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*. Springer-Verlag, 2006.
- M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. *ICML*, 2016.
- H. He, G. Huang, and Y. Yuan. Asymmetric valleys: Beyond sharp and flat local minima. *NeurIPS*, 2019.
- S. Hochreiter and J. Schmidhuber. Flat minima. *Neural Computation*, 9, 1997.
- P. Izmailov, D. Podoprikin, T. Garipov, D. Vetrov, and A.G. Wilson. Averaging weights leads to wider optima and better generalization. *UAI*, 2018.

- A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: convergence and generalization in neural networks. *NeurIPS*, 2018.
- N.S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P.T.P. Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *ICLR*, 2017.
- J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3), 1952.
- J. Langford and M. Seeger. Bounds for averaging classifiers. *CMU technical report*, 2001.
- J. Lee, L. Xiao, S.S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. *Journal of Statistical Mechanics*, 2020.
- G. Letarte, P. Germain, B. Guedj, and F. Laviolette. Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks. *NeurIPS*, 2019.
- G. Lugosi and G. Neu. Generalization bounds via convex analysis. *COLT*, 2022.
- A. Maurer. A note on the PAC Bayesian theorem. *arXiv:0411099*, 2004.
- D.A. McAllester. Some PAC-Bayesian theorems. *COLT*, 1998.
- S. Mei and A. Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4), 2022.
- K. Miyaguchi. PAC-Bayesian transportation bound. *arXiv:1905.13435*, 2019.
- W. Mou, L. Wang, X. Zhai, and K. Zheng. Generalization bounds of SGLD for non-convex learning: Two theoretical viewpoints. *COLT*, 2018.
- V. Nagarajan and J.Z. Kolter. Deterministic PAC-Bayesian generalization bounds for deep networks via generalizing noise-resilience. *ICLR*, 2019.
- J. Negrea, M. Haghifam, G.K. Dziugaite, A. Khisti, and D.M. Roy. Information-Theoretic generalization bounds for SGLD via Data-Dependent estimates. *NeurIPS*, 2019.
- G. Neu, G.K. Dziugaite, M. Haghifam, and D.M. Roy. Information-theoretic generalization bounds for stochastic gradient descent. *COLT*, 2021.
- B. Neyshabur, S. Bhojanapalli, and N. Srebro. A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. *ICLR*, 2018.
- G.A. Pavliotis. *Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations*. Springer, 2014.
- A. Pensia, V. Jog, and P.L. Loh. Generalization error bounds for noisy, iterative algorithms. *IEEE International Symposium on Information Theory*, 2018.

- M. Pérez-Ortiz, O. Risvaplata, J. Shawe-Taylor, and C. Szepesvári. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22, 2021a.
- M. Pérez-Ortiz, O. Rivasplata, B. Guedj, M. Gleeson, J. Zhang, J. Shawe-Taylor, M. Bober, and J. Kittler. Learning PAC-Bayes priors for probabilistic neural networks. *arXiv:2109.10304*, 2021b.
- N. Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1), 1999.
- M. Raginsky, A. Rakhlin, and M. Telgarsky. Non-convex learning via stochastic gradient Langevin dynamics: a nonasymptotic analysis. *COLT*, 2017.
- A. Rahimi and B. Recht. Random features for large-scale kernel machines. *NeurIPS*, 2007.
- A. Raj, L. Zhu, M. Gürbüzbalaban, and U. Şimşekli. Algorithmic stability of Heavy-Tailed SGD with general loss functions. *arXiv:2301.11885*, 2023.
- O. Rivasplata, I. Kuzborskij, C. Szepesvári, and J. Shawe-Taylor. PAC-Bayes analysis beyond the usual bounds. *NeurIPS*, 2020.
- D. Russo and J. Zou. How much does your data exploration overfit? Controlling bias via information usage. *IEEE Transactions on Information Theory*, 66(1), 2019.
- L. Sagun, U. Evci, V.U. Güney, Y. Dauphin, and L. Bottou. Empirical analysis of the Hessian of over-parametrized neural networks. *ICLR Workshop*, 2018.
- A.R. Sankar, Y. Khasbage, R. Vigneswaran, and V.N. Balasubramanian. A deeper look at the Hessian eigenspectrum of deep neural networks and its applications to regularization. *AAAI Conference on Artificial Intelligence*, 2020.
- S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.
- Y. Song, J. Sohl-Dickstein, D.P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *ICLR*, 2021.
- I.O. Tolstikhin and Y. Seldin. PAC-Bayes-empirical-Bernstein inequality. *NeurIPS*, 2013.
- V.N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.
- P. Viallard, P. Germain, A. Habrard, and E. Morvant. A general framework for the disintegration of PAC-Bayesian bounds. *arXiv:2102.08649*, 2021.
- A. Xu and M. Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. *NeurIPS*, 2017.
- G. Yang and E. Littwin. Tensor programs IIb: Architectural universality of neural tangent kernel training dynamics. *ICML*, 2021.
- V. Zantedeschi, P. Viallard, E. Morvant, R. Emonet, A. Habrard, P. Germain, and B. Guedj. Learning stochastic majority votes by minimizing a PAC-Bayes generalization bound. *NeurIPS*, 2021.

- C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *Communications of the ACM*, 64, 11 2016.
- W. Zhou, V. Veitch, M. Austern, R.P. Adams, and P. Orbanz. Non-vacuous generalization bounds at the ImageNet scale: a PAC-Bayesian compression approach. *ICLR*, 2019.

Appendix A. Omitted proofs

Corollary 2 *If $\ell(h, X)$ is R -sub-Gaussian for each $h \in \mathcal{H}$, the bound (2) takes the form*

$$\mathcal{L}_{\mathcal{Z}}(h_T) \leq \mathcal{L}_s(h_T) + \frac{1}{\sqrt{m}} \left(\log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \int_0^T \Delta \mathcal{C}_s(h_t) dt + \log \frac{1}{\delta} + \frac{R^2}{2} \right).$$

Moreover, if ℓ is bounded in $[0, 1]$ we have the tighter bound

$$\mathcal{L}_{\mathcal{Z}}(h_T) \leq \text{kl}^{-1} \left(\mathcal{L}_s(h_T) \left| \frac{\log \frac{\rho_0(h_0)}{\rho_0(h_T)} + \log \frac{2\sqrt{m}}{\delta} + \int_0^T \Delta \mathcal{C}_s(h_t) dt}{m} \right. \right).$$

Proof The first bound follows from Theorem 1 with $\Psi(u, v) = \sqrt{m}(v - u)$. In this case $\xi \leq R^2/(2m)$ by the definition of sub-Gaussianity. The second bound follows from Theorem 1 with $\Psi(u, v) = m \text{kl}(u||v)$, after using the fact that with this choice one has $\xi \leq 2\sqrt{m}$ if the loss is bounded in $[0, 1]$ (Bégin et al., 2016). ■

Theorem 3 *Let $K \in \mathbb{N}$ and $\delta \in (0, 1)$, and let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable map. For each dataset $s \in \mathcal{Z}^m$, denote as A_s a Borel set on which \mathcal{C}_s is twice-differentiable and M -smooth, with $\sup_k \eta_k \leq 1/(2M)$. For $(s, h_0) \sim \mu^m \otimes \rho_0$, assume that the trajectory $\{h_k\}_{k=0}^{K-1}$ lies in A_s with probability higher than $1 - \delta/2$. Then, with a probability of at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$,*

$$\Psi(\mathcal{L}_s(h_T), \mathcal{L}_{\mathcal{Z}}(h_T)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_k)} - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) + \log \frac{2\xi}{\delta} + \delta/2,$$

where $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_s(h), \mathcal{L}_{\mathcal{Z}}(h))} d\mu^m(\bar{s}) d\rho_0(h)$.

Proof First, for any $s \in \mathcal{Z}^m$ define

$$A_s^0 = \{h_0 \in \mathcal{H} : h_k \in A_s \text{ for all } k \in \{0, \dots, K-1\}\},$$

which is a Borel set thanks to the regularity of G on A_s .

We start by noticing that for all k , the restriction of G_{η_k} to A_s is injective. Indeed, if h and h' are in A_s , we have that

$$\|G_{\eta_k}(h) - G_{\eta_k}(h')\| \geq \|h - h'\| - \eta_k \|\nabla \mathcal{C}_s(h) - \nabla \mathcal{C}_s(h')\| \geq \frac{1}{2} \|h - h'\|.$$

For any fixed s , if we condition on $h_0 \in A_s^0$, we have that this condition is satisfied for all $k \in \{0, \dots, K-1\}$. Now let $\tilde{\rho}_k^s = \rho_k(\cdot | h_0 \in A_s^0)$ be the law of h_k , conditioned on $h_0 \in A_s^0$. If we denote as \tilde{G}_k the restriction of G_{η_k} to $\text{supp } \tilde{\rho}_k^s$, we have that \tilde{G}_k is a differentiable bijection $\text{supp } \tilde{\rho}_k^s \leftrightarrow \text{supp } \tilde{\rho}_{k+1}^s$. In particular, we see by induction that this implies that $\tilde{\rho}_{k+1}^s$ admits a Lebesgue density (since $\tilde{\rho}_0^s \ll \rho_0$), and by the change of variable formula

$$\tilde{\rho}_{k+1}^s(h) = \tilde{\rho}_k^s \circ \tilde{G}_k^{-1}(h) \det \left[\text{Id} - \eta_k \nabla^2 \mathcal{C}_s \circ \tilde{G}_k^{-1}(h) \right]^{-1}, \quad \forall h \in \text{supp } \tilde{\rho}_{k+1}^s.$$

Since $\tilde{G}_k(h_k) = h_{k+1}$ for $h_k \in A_s$, we get

$$\tilde{\rho}_{k+1}^s(h_{k+1}) = \tilde{\rho}_k^s(h_k) \det \left[\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right]^{-1}.$$

In particular, we have that for $h_0 \in A_s^0$

$$\begin{aligned} \log \frac{\tilde{\rho}_K^s(h_K)}{\tilde{\rho}_0^s(h_0)} &= \sum_{k=0}^{K-1} \log \frac{\tilde{\rho}_{k+1}^s(h_{k+1})}{\tilde{\rho}_k^s(h_k)} \\ &= - \sum_{k=0}^{K-1} \log \det \left[\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right] = - \sum_{k=0}^{K-1} \text{tr} \log \left[\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right]. \end{aligned} \quad (10)$$

where the last equality follows from the Jacobi formula for positive definite matrices, namely $\log \det = \text{tr} \log$.

We now use the same Markov argument as in Theorem 1, with posterior $\tilde{\rho}_K^s$ and prior ρ_0 . Explicitly, we have that with probability at least $1 - \frac{\delta}{2}$ on $(s, h_K) \sim \mu^m * \tilde{\rho}_K$

$$e^{\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) - \log \frac{\tilde{\rho}_K^s(h_K)}{\rho_0(h_K)}} \leq \frac{2}{\delta} \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h)) - \log \frac{\tilde{\rho}_K^{\bar{s}}(h)}{\rho_0(h)}} d\mu^m(\bar{s}) d\tilde{\rho}_K^{\bar{s}}(h).$$

Since for every $\bar{s} \in \mathcal{Z}^m$ we get

$$\begin{aligned} &\int_{\mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h)) - \log \frac{\tilde{\rho}_K^{\bar{s}}(h)}{\rho_0(h)}} d\tilde{\rho}_K^{\bar{s}}(h) \\ &= \int_{\{\tilde{\rho}_k > 0\}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h)) - \log \frac{\tilde{\rho}_K^{\bar{s}}(h)}{\rho_0(h)}} d\tilde{\rho}_K^{\bar{s}}(h) \leq \int_{\mathcal{H}} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h))} d\rho_0(h), \end{aligned}$$

we get that

$$\mu^m * \rho_K \left(\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) \leq \log \frac{\tilde{\rho}_K^s(h_K)}{\rho_0(h_K)} + \log \frac{2\xi}{\delta} \middle| h_0 \in A_s^0 \right) \geq 1 - \delta/2.$$

Now, we note that for any $h_0 \in A_s^0$, the following holds:

$$\log \frac{\tilde{\rho}_k^s(h_K)}{\rho_0(h_K)} = \log \frac{\tilde{\rho}_k^s(h_K)}{\tilde{\rho}_0^s(h_0)} + \log \frac{\rho_0^s(h_0)}{\rho_0(h_K)} - \log \rho_0(A_s^0),$$

which is further bounded noticing that $-\log(1 - \delta/2) \leq \delta/2$. In particular, using the change of density formula (10) we get that

$$\begin{aligned} &\mu^m * \rho_K \left(\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) \right. \\ &\quad \left. \leq \log \frac{\rho_0(h_0)}{\rho_0(h_K)} - \sum_{k=0}^{K-1} \text{tr} \log \left[\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right] + \log \frac{2\xi}{\delta} + \frac{\delta}{2} \middle| h_0 \in A_s^0 \right) \geq 1 - \delta/2. \end{aligned} \quad (11)$$

Now, note that for any event E we have

$$\begin{aligned}\mu^m * \rho_K(E) &= \int_{\mathcal{Z}^m} (\rho_K(E|h_0 \in A_s^0)\rho_0(A_s^0) + \mu^m * \rho_K(E|h_0 \in A_s^0)(1 - \rho_0(A_s^0))) d\mu^m(s) \\ &\leq \mu^m * \rho_K(E|h_0 \in A_s^0) + \frac{\delta}{2},\end{aligned}$$

since $\mu^m \otimes \rho_0(h_0 \notin A_0^s) \leq \delta/2$ by hypothesis. Applying this to (11), we get that

$$\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_K)} - \sum_{k=0}^{K-1} \text{tr} \log \left[\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right] + \log \frac{2\xi}{\delta} + \frac{\delta}{2}, \quad (12)$$

with probability at least $1 - \delta$ on $(s, h_K) \sim \mu^m * \rho_K$. Since sampling from ρ_K for a given s is equivalent to sample from ρ_0 and follow the dynamics until the K -th step, we can claim that (12) holds with probability at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$. ■

Lemma 4 *With the notation of Theorem 3, let $h_k \in A_s$. Then*

$$- \text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) \leq \eta_k \Delta \mathcal{C}_s(h_k) + \eta_k^2 \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{F}}^2 \leq \frac{3}{2} \eta_k \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{TR}},$$

where $\|\cdot\|_{\text{F}}$ is the Frobenius norm and $\|\cdot\|_{\text{TR}}$ the trace norm.

Proof To obtain the first upper bound, denote as $\{\lambda_i(h_k)\}_{i=1}^d$ the spectrum of $\nabla^2 \mathcal{C}_s(h_k)$. Then we have that

$$- \text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) = - \sum_{k=0}^{K-1} \sum_{i=1}^d \log(1 - \eta_k \lambda_i(h_k)).$$

Using that $-\log(1 - u) \leq u(u + 1)$ for $|u| \leq 1/2$, we obtain that for each k

$$- \sum_{i=1}^d \log(1 - \eta_k \lambda_i(h_k)) \leq \eta_k \sum_{i=1}^d \lambda_i(h_k) + \eta_k^2 \sum_{i=1}^d \lambda_i(h_k)^2 = \eta_k \Delta \mathcal{C}_s(h_k) + \eta_k^2 \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{F}}^2.$$

For the second upper bound, note that $\Delta \mathcal{C}_s(h_k) \leq \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{TR}}$ and

$$\eta_k^2 \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{F}}^2 = \eta_k^2 \sum_{k=1}^d \lambda_i(h_k)^2 \leq \eta_k \sum_{k=1}^d |\eta_k \lambda_i(h_k)| |\lambda_i(h_k)| = \frac{1}{2} \eta_k \|\nabla^2 \mathcal{C}_s(h_k)\|_{\text{TR}},$$

where we used that $|\eta_k \lambda_i(h_k)| \leq \eta_k \|\nabla^2 \mathcal{C}_s(h_k)\| \leq 1/2$ since \mathcal{C}_s is $1/(2\eta_k)$ -smooth in $h_k \in A_s$. ■

Appendix B. Upper bounding kl^{-1}

Recall that $\text{kl}(u||v) = u \log \frac{u}{v} + (1 - u) \log \frac{1-u}{1-v}$ is well defined on $(0, 1)^2$ and has range $(0, 1)$. If we restrict to the region $\{v \geq u\}$, then the function is injective and we can define the inverse $\text{kl}^{-1}(u|c)$, which returns the only $v \geq u$ such that $\text{kl}(u||v) = c$. Here, we upper bound this inverse.

Lemma 7 $\text{kl}^{-1}(u|c) \leq \min\{2(u+c), u + \sqrt{c/2}\}$.

Proof It is well known in the literature that

$$\text{kl}^{-1}(u|c) \leq u + \sqrt{c/2}$$

(see for instance [Dziugaite and Roy \(2017\)](#)). Now we want to show that $\text{kl}^{-1}(u|c) \leq 2(u+c)$. Fix $u \in (0, 1)$ and define on $(u, 1)$ the map

$$h_u : v \mapsto \text{kl}(u|v).$$

We have that $h'_u(v) = \frac{1-u}{1-v} - \frac{u}{v}$ and $h''_u(v) = \frac{1-u}{(1-v)^2} + \frac{u}{v^2}$. In particular, h_u is convex. Define the straight line

$$\sigma_u : v \mapsto (v - 2u)/2.$$

We want to show that $h_u(v) \geq \sigma_u(v)$ for all $v \in (u, 1)$. Indeed, if this is the case, then $h_u^{-1}(c) \leq \sigma_u^{-1}(c) = 2(u+c)$ and the statement of the lemma follows. Since h_u is convex, it is enough to show that σ_u lies below the tangent of h_u with slope $1/2$, which we denote as τ_u . Let v_u be the value of v such that $h'_u(v_u) = 1/2$ (which always exists in $(0, 1)$ since $h'_u(v) \rightarrow 0$ for $v \rightarrow u$ and $h'_u(v) \rightarrow \infty$ for $v \rightarrow 1$). We have that $v_u = \frac{1}{2}(\sqrt{1+8u} - 1)$. Now, note that $u \mapsto v_u$ is strictly concave, increasing in $(0, 1)$, and tends to 0 for $u \rightarrow 0$ and to 1 for $u \rightarrow 1$. In particular, we have that $v_u > u$ for all u . We hence know that $\text{kl}(u|v_u)$ is well defined and positive. This means that $\tau_u(v_u) > 0$. On the other hand, we have that

$$u \mapsto \sigma_u(v_u) = \frac{1}{4}(\sqrt{1+8u} - 1) - u$$

is a strictly concave function that vanishes in 0. As its derivative for $u \rightarrow 0$ tends to 0, we have $\sigma_u(v_u) < 0$ for all $u \in (0, 1)$. In particular $\tau_u(v_u) < \sigma_u(v_u)$. Since τ_u and σ_u have the same slope, we get that σ_u lies below τ_u and so we conclude. \blacksquare

Appendix C. Rewriting the Laplacian's integral

We explicitly derive here (4). First, notice that $\partial_t \log \|\nabla \mathcal{C}_s(h_t)\| = -\frac{\nabla \mathcal{C}_s(h_t)}{\|\nabla \mathcal{C}_s(h_t)\|} \cdot \nabla \|\nabla \mathcal{C}_s(h_t)\|$. Since, for all h ,

$$\Delta \mathcal{C}_s(h) = \nabla \cdot \nabla \mathcal{C}_s(h) = \nabla \cdot \left(\frac{\nabla \mathcal{C}_s(h)}{\|\nabla \mathcal{C}_s(h)\|} \right) \|\nabla \mathcal{C}_s(h)\| + \frac{\nabla \mathcal{C}_s(h)}{\|\nabla \mathcal{C}_s(h)\|} \cdot \nabla \|\nabla \mathcal{C}_s(h)\|,$$

we get

$$\Delta \mathcal{C}_s(h_t) = \nabla \cdot \left(\frac{\nabla \mathcal{C}_s(h_t)}{\|\nabla \mathcal{C}_s(h_t)\|} \right) \|\nabla \mathcal{C}_s(h_t)\| - \partial_t \log \|\nabla \mathcal{C}_s(h_t)\|.$$

We conclude that

$$\int_0^T \Delta \mathcal{C}_s(h_t) dt = \log \frac{\|\nabla \mathcal{C}_s(h_0)\|}{\|\nabla \mathcal{C}_s(h_T)\|} + \int_0^T \nabla \cdot \left(\frac{\nabla \mathcal{C}_s(h_t)}{\|\nabla \mathcal{C}_s(h_t)\|} \right) \|\nabla \mathcal{C}_s(h_t)\| dt.$$

The integral in the RHS is a line-integral along the path $h_{[0,T]}$, as $\|\nabla\mathcal{C}_s(h)\|$ is the norm of the flow’s “velocity” in h . Moreover, $\tau(h) = -\frac{\nabla\mathcal{C}_s(h)}{\|\nabla\mathcal{C}_s(h)\|}$ is the unit tangent vector to the gradient flow in h . We can thus write

$$\int_0^T \Delta\mathcal{C}_s(h_t)dt = \log \frac{\|\nabla\mathcal{C}_s(h_0)\|}{\|\nabla\mathcal{C}_s(h_T)\|} - \int_{h_{[0,T]}} \nabla \cdot \tau(h) \|\delta h\|,$$

which is (4).

Appendix D. Examples

D.1. Calculations for the random feature model

We derive here the formula for $\Delta\mathcal{C}_s$ for the random feature model of Section 5.1. We let

$$F(x) = \frac{1}{\sqrt{d}}h\Phi(x),$$

where the h is a $q \times d$ learnable matrix and $\Phi : \mathbb{R}^p \rightarrow \mathbb{R}^d$ is a fixed feature map, chosen randomly at initialisation.

In general, for a generic model with a twice differentiable $\hat{\ell}$ one has that

$$\Delta\hat{\ell} = \sum_i \frac{\partial\hat{\ell}}{\partial F^i} \Delta F^i + \sum_{ii'} \frac{\partial^2\hat{\ell}}{\partial F^i \partial F^{i'}} \nabla F^i \cdot \nabla F^{i'},$$

which follows easily from the chain rule. Here the model is linear in h and so we are simply left with

$$\Delta\hat{\ell} = \frac{1}{d}\|\Phi\|^2 \Delta_F \hat{\ell},$$

where $\Delta_F \hat{\ell} = \sum_i \partial_{F^i}^2 \hat{\ell}$.

When $\hat{\ell}$ is the cross entropy loss, we get

$$\Delta_F \hat{\ell} = \sum_i \partial_{F^i}^2 \hat{\ell} = \sum_i \left(\frac{e^{F^i}}{\sum_j e^{F^j}} - \frac{e^{2F^i}}{(\sum_j e^{F^j})^2} \right) = 1 - \frac{\sum_i e^{2F^i}}{(\sum_i e^{F^i})^2} \leq 1,$$

and so

$$\|\nabla^2\mathcal{C}_s(h)\|_{\text{TR}} = \Delta\mathcal{C}_s(h) = \frac{1}{m} \sum_{z \in s} \frac{1}{d} \|\Phi(x)\|^2 \left(1 - \frac{\sum_i e^{2F^i(x)}}{(\sum_i e^{F^i(x)})^2} \right) \leq \frac{1}{m} \sum_{z \in s} \frac{1}{d} \|\Phi(x)\|^2.$$

D.2. Wide Neural Networks

We consider a fully connected neural network $F_h : \mathbb{R}^{n_0} \rightarrow \mathbb{R}$, for some $n_0 \in \mathbb{N}$. For simplicity, we let each hidden layer have identical width $n \in \mathbb{N}$ and activation function $\phi : \mathbb{R} \rightarrow \mathbb{R}$. We assume that the inputs are coming from a compact set $\mathcal{X} \subset \mathbb{R}^{n_0}$.

The network output is determined by an input $x^0 \in \mathcal{X}$, weights $\{W^l\}_{l=2}^{L-1} \subset \mathbb{R}^{n \times n}$, $W^1 \in \mathbb{R}^{n_0 \times n}$ and $W^L \in \mathbb{R}^{n \times 1}$, and biases $\{b^l\}_{l=1}^{L-1} \subset \mathbb{R}^n$ and $b^L \in \mathbb{R}$. We use h to denote the vector of all weights and biases. The network’s output is

$$F_h(x^0) = x^{L-1}W^L + b^L,$$

where we define

$$x^{l+1} = \phi(x^l W^{l+1} + b^{l+1}), \quad \text{for } l = 0, \dots, L-2,$$

where ϕ is applied component-wise. We consider a dataset $s = (x_i, y_i)_{i=1}^m \in \mathcal{Z}^m$ sampled from the measure μ^m . We consider a square loss objective in the form

$$\mathcal{C}_s(h) = \frac{1}{m} \sum_{i=1}^m (F_h(x_i) - y_i)^2.$$

We consider a Gaussian initialisation ρ_0 , where the all of the parameters are independently drawn as

$$W_{ij}^l \sim \mathcal{N}\left(0, \frac{\sigma_w^2}{n}\right), \quad b_i^l \sim \mathcal{N}(0, \sigma_b^2),$$

for some positive σ_w and σ_b .

If the training step-size is scaled appropriately, the large width limit is known to reduce to the neural tangent kernel (NTK) dynamics (Jacot et al., 2018). Given $x, x' \in \mathcal{X}$, the value of the NTK $\Theta(x, x') \in \mathbb{R}$ is given by the limit (in probability) as $n \rightarrow \infty$ of the quantity

$$\hat{\Theta}(x, x') = \frac{1}{n} \langle \nabla F_{h_0}(x), \nabla F_{h_0}(x') \rangle,$$

with $h_0 \sim \rho_0$. We borrow the analysis of Lee et al. (2020), who use this fact to study the convergence on the finite width NN under training with gradient descent. To leverage ideas from this analysis, we must make the following additional assumptions:

1. The analytic NTK Θ is full-rank with minimum and maximum eigenvalues satisfying $0 < \lambda_{\min} \leq \lambda_{\max} < \infty$.
2. $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ is compact and the data distribution μ has no atoms.
3. The activation function ϕ has Lipschitz continuous and bounded gradients.

Lemma 8 (Lemma 1, Lee et al. (2020)) *Suppose assumptions 1-3 are satisfied, then there exists a constant $R > 0$ such that, for any fixed $C > 0$ and $\delta \in (0, 1)$, for n sufficiently large we can find a convex subset $A(C, \delta, n) \subseteq \mathcal{H}$, such that $\rho_0(A(C, \delta, n)) \geq 1 - \delta/2$ and*

$$\|\nabla F_h(x) - \nabla F_{h'}(x)\|_{\mathbb{F}} \leq \sqrt{n}R \|h - h'\|; \quad \|\nabla F_h(x)\|_{\mathbb{F}} \leq \sqrt{n}R,$$

for all $h, h' \in B(h_0, Cn^{-1/2})$, $h_0 \in A(C, \delta, n)$ and $x \in \mathcal{X}$.

Here we use the notation $B(h, r)$ to denote the ball about h of radius r . As usual, the notation ∇ denote derivatives with respect to the parameters. The result is not stated so explicitly in the paper of Lee et al. (2020), but can be deduced easily by following the proof therein. The convexity of the set $A(C, \delta, n)$ follows from its construction by upper bounding the operator norms of the weight matrices. Similarly, we state more formally another result from this work.

Lemma 9 (Theorem G.1, Lee et al. (2020)) *Suppose assumptions 1-3 are satisfied, $s \in \text{supp}(\mu^m)$ and let $\eta_{\star} = 2(\lambda_{\min} + \lambda_{\max})^{-1}$. Then there exists constants $C, R_0 > 0$ such that for sufficiently large n , whenever $\sup_k \eta_k < \eta_{\star}/n$ and $h_0 \in A(C, \delta, n)$,*

$$\|h_k - h_0\| \leq Cn^{-1/2}, \quad \mathcal{C}_s(h_k) \leq R_0, \quad \text{for each } k \in \mathbb{N}.$$

Now we state Proposition 5 more formally.

Proposition 5 (Rigorous statement) *Suppose assumptions 1-3 are satisfied and let $\eta_\star = 2(\lambda_{\min} + \lambda_{\max})^{-1}$. Then, there exists positive constants C , R , and R_0 such that, for any $\delta \in (0, 1)$ and sufficiently large n , whenever $\sup_k \eta_k < \frac{1}{n} \min\{\eta_\star, (2R(R + R_0^{1/2}))^{-1}\}$, with a probability of at least $1 - \delta$ on $(s, h_0) \sim \mu^m \otimes \rho_0$*

$$\Psi(\mathcal{L}_s(h_T), \mathcal{L}_Z(h_T)) \leq \frac{C}{\sigma_w^2} \left(\|h_0\|n^{1/2} + C/2 \right) - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h_k) \right) + \log \frac{2\xi}{\delta} + \delta/2,$$

where $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_s(h), \mathcal{L}_Z(h))} d\mu^m(s) d\rho_0(h)$.

Proof Let C be the constant given in Lemma 9 and suppose n is sufficiently large so that its conclusion is satisfied. We construct the set A_s as the set containing all points visited by gradient flows starting in $A(C, \delta, n)$. By construction, the entire trajectory $\{h_k\}_{k=0}^K$ lies in this set with probability at least $1 - \delta$. To show that \mathcal{C}_s is smooth on this set, we note that for any $h, h' \in A_s(C, \delta, n)$,

$$\begin{aligned} & \|\nabla \mathcal{C}_s(h) - \nabla \mathcal{C}_s(h')\| \\ & \leq \frac{1}{m} \sum_{i=1}^m (|F_h(x_i) - y_i| \|\nabla F_h(x_i) - \nabla F_{h'}(x_i)\| + \|\nabla_h F_{h'}(x_i)\| \|F_h(x_i) - F_{h'}(x_i)\|) \\ & \leq \frac{1}{m} \sum_{i=1}^m (|F_h(x_i) - y_i| \|\nabla F_h(x_i) - \nabla F_{h'}(x_i)\| + \|\nabla F_{h'}(x_i)\| \|\nabla F_{\hat{h}}(x_i)\| \|h - h'\|), \end{aligned}$$

for some $\hat{h} \in \{\tau h + (1 - \tau)h' : \tau \in [0, 1]\}$.

From Lemma 8, it follows that $\|\nabla F_{h'}(x_i)\|_{\mathbb{F}} \leq \sqrt{n}R$. In fact, this holds for all parameter values of distance $Cn^{-1/2}$ from $A(C, \delta, n)$. Since this is a convex set which both h and h' belong to, \hat{h} must belong to this set also, and so $\|\nabla F_{\hat{h}}(x_i)\|_{\mathbb{F}} \leq \sqrt{n}R$. Additionally, we apply Lemma 8 as well as the Cauchy-Schwarz inequality to deduce,

$$\frac{1}{m} \sum_{i=1}^m |F_h(x_i) - y_i| \|\nabla F_h(x_i) - \nabla F_{h'}(x_i)\| \leq \sqrt{n}R \mathcal{C}_s(h)^{1/2} \|h - h'\| \leq \sqrt{n}R R_0^{1/2} \|h - h'\|$$

From this, we deduce that

$$\|\nabla \mathcal{C}_s(h) - \nabla \mathcal{C}_s(\tilde{h})\| \leq (R_0^{1/2} R \sqrt{n} + R^2 n) \|h - h'\| \leq R(R_0^{1/2} + R)n \|h - h'\|,$$

which means that the optimisation objective is M -smooth on $A_s(C, \delta, n)$, with $M = R(R_0^{1/2} + R)n$.

Since, by hypothesis, we consider a schedule such that $\sup_k \eta_k \leq 1/(2M)$, Theorem 3 applies. Moreover, we have

$$\log \frac{\rho_0(h_0)}{\rho_0(h_k)} \leq \frac{n}{2\sigma_w^2} \left| \|h_k\|^2 - \|h_0\|^2 \right| \leq \frac{n}{2\sigma_w^2} \|h_k - h_0\| (\|h_k - h_0\| + 2\|h_0\|).$$

Thus, using Lemma 9,

$$\log \frac{\rho_0(h_0)}{\rho_0(h_k)} \leq \frac{C}{\sigma_w^2} (C/2 + \|h_0\|n^{1/2}),$$

from which the statement follows. ■

As noted in [Lee et al. \(2020\)](#) and [Yang and Littwin \(2021\)](#), the NTK analysis of wide neural networks can be performed in settings where the hidden layers have different widths and on networks with different architectures. Therefore, we should expect a similar argument to that given above can be reproduced in these settings.

Appendix E. The analysis for general iterative methods

The proof of [Theorem 6](#) follows immediately from the proof technique of [Theorem 3](#) as soon as $-\nabla\mathcal{C}_s$ is replaced by the vector field. Similarly, we obtain the following result for the continuous-time flow dynamics.

Theorem 10 *Consider the dynamics $\partial_t h_t = V_s(h_t; t)$, with $V_s : \mathcal{H} \times \mathbb{R}^+ \rightarrow \mathcal{H}$ differentiable in h . Let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable function. For any $\delta \in (0, 1)$ and $T > 0$, with probability at least $1 - \delta$ on the random draw $(s, h_0) \sim \mu^m \otimes \rho_0$, we have*

$$\Psi(\mathcal{L}_s(h_T), \mathcal{L}_Z(h_T)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_T)} - \int_0^T \nabla \cdot V_s(h_t; t) dt + \log \frac{\xi}{\delta},$$

where ∇ refers to derivatives with respect to h and $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}} e^{\Psi(\mathcal{L}_s(h), \mathcal{L}_Z(h))} d\mu^m(s) d\rho_0(h)$.

E.1. Mini-batches

We can consider here a version of gradient descent that only evaluates the training objective on a mini-batch $s_k \subset s$ at each iteration $k \in \mathbb{N}$, as already discussed in [Section 6](#). Note that the choice of the mini-batch can be random and doesn't need to be known a priori. Our result will always apply on the specific realisation of the sequence of batches used for the training.

We consider an objective in the form

$$\mathcal{C}_s(h) = \frac{1}{m} \sum_{z \in s} \hat{\ell}(z, h),$$

for some surrogate loss function $\hat{\ell} : \mathcal{Z} \times \mathcal{H} \rightarrow \mathbb{R}$. For a batch $s_k \subset s$ of m_k elements, we write

$$\mathcal{C}_{s_k}(h) = \frac{1}{m_k} \sum_{z \in s_k} \hat{\ell}(z, h).$$

Proposition 11 *For a sequence of batches $\{s_k\}$, consider the dynamics $h_{k+1} = h_k - \eta_k \mathcal{C}_{s_k}(h_k)$. We denote as $\tilde{\mu}^m$ the law of $(s, \{s_k\})$, which takes into account the potential randomness in the choice of the batches. For each k , let A_s be a Borel where $\hat{\ell}(z, \cdot)$ is twice differentiable and M -smooth for every z in s , with $\max_k \eta_k \leq 1/(2M)$. Let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable function. Fix $K \in \mathbb{N}$ and let $\delta \in (0, 1)$, such that the trajectory $\{h_k\}_{k=0}^{K-1}$ lies in A_s , with probability at least $1 - \delta/2$ on the randomness of the training dataset s , the initialisation h_0 , and the choice of the batches. With probability at least $1 - \delta$ on the same randomness, we have*

$$\Psi(\mathcal{L}_s(h_K), \mathcal{L}_Z(h_K)) \leq \log \frac{\rho_0(h_0)}{\rho_0(h_K)} - \sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} + \eta_k \nabla^2 \mathcal{C}_{s_k}(h_k) \right) + \log \frac{2\xi}{\delta} + \frac{\delta}{2},$$

with $\xi = \int_{\mathcal{Z} \times \mathcal{H}} e^{\Psi(\mathcal{L}_{\tilde{s}}(h), \mathcal{L}_Z(h))} d\mu^m(\tilde{s}) d\rho_0(h)$.

E.2. Momentum

We can consider the use of auxiliary variables: instead of having just h_k , we take the pair of processes (h_k, v_k) . If the updates of these processes are of the form

$$\begin{pmatrix} h_{k+1} \\ v_{k+1} \end{pmatrix} = \begin{pmatrix} h_k \\ v_k \end{pmatrix} + V_s(h_k, v_k; k),$$

for some iteration-dependent vector field V_s , then the usual analysis applies immediately. For example, let us consider the momentum scheme

$$h_{k+1} = h_k + v_{k+1}, \quad v_{k+1} = \mu_k v_k - \eta_k \nabla \mathcal{C}_s(h_k),$$

where $\mu_k \in [0, 1]$ is the momentum schedule.

This corresponds to the vector field

$$V_s(h, v; k) = \begin{pmatrix} \mu_k v - \eta_k \nabla \mathcal{C}_s(h) \\ (\mu_k - 1)v - \eta_k \nabla \mathcal{C}_s(h) \end{pmatrix},$$

whose Jacobian reads

$$\nabla V_s(h, v; k) = \begin{pmatrix} -\eta_k \nabla^2 \mathcal{C}_s(h) & \mu_k \text{Id} \\ -\eta_k \nabla^2 \mathcal{C}_s(h) & (\mu_k - 1) \text{Id} \end{pmatrix}.$$

We can easily compute

$$\det(\text{Id} + \nabla V_s(h, v; k)) = \det\left(\mu_k (\text{Id} - \eta_k \nabla^2 \mathcal{C}_s(h)) + \mu_k \eta_k \nabla^2 \mathcal{C}_s(h)\right) = \det(\mu_k \text{Id}) = \mu_k^d,$$

and so

$$-\sum_{k=0}^{K-1} \text{tr} \log \left(\text{Id} + \nabla V_s(h_k, v_k; k) \right) = -\sum_{k=0}^{K-1} \log \det \left(\text{Id} + \nabla V_s(h_k, v_k; k) \right) = d \sum_{k=0}^{K-1} \log \frac{1}{\mu_k}.$$

If we consider the schedule $\mu_k = 1 - \alpha(k+1)^{-1}$, for some fixed $\alpha < 1$, then we obtain that this sum scales with $\alpha d \log K$, and this is made dimension independent by choosing $\alpha \sim 1/d$. Curiously, when $\mu_k \equiv 1$, the sum vanishes. As a final remark, note that one must initialise the pair (h_0, v_0) by drawing it from a fixed distribution ρ_0 , that we assume to have full support on $\mathcal{H} \times \mathbb{R}^d$. This excludes the case of a deterministic initial value for the velocity.

Appendix F. Discretised damped Hamiltonian dynamics

Here, we consider a Hamiltonian approach, and hence we introduce d additional variables v , representing the velocities (momenta) of the parameters h . The idea is to exploit the fact that the joint density of the pair $(h, v) \in \mathcal{H}^2$ is conserved under the Hamiltonian flow, a property that is preserved for discrete time-steps by suitable symplectic integrators (Hairer et al., 2006). In order to solve an optimisation problem, we can alternate conservative Hamiltonian steps with dissipative ones, which only involve v and entail an exactly computable change in density.

Let us make things more concrete. For the rest of this section, we denote as ρ_k the joint density of (h_k, v_k) . We consider an increasing differentiable map $\psi : \mathbb{R} \rightarrow \mathbb{R}$, such that $\psi(0) = 0$, and

we fix $\eta > 0$. We denote as $\Psi_\eta(v)$ the value at $t = \eta$ of the solution of $\partial_t \tilde{v}_t = -\psi(\tilde{v}_t)$ satisfying $\tilde{v}_0 = v_k$, where with a slight abuse of notation we are here implying that ψ is acting component-wise on $\tilde{v}_t \in \mathbb{R}^d$. From (h_k, v_k) , to evaluate (h_{k+1}, v_{k+1}) we first proceed with a dissipative step:

$$h_{k+1/2} = h_k; \quad v_{k+1/2} = \Psi_\eta(v_k).$$

Since this step involves the exact solution of a continuous-time gradient descent evolution, we can appeal to the usual continuity arguments to show that

$$\log \frac{\rho_{k+1/2}(h_{k+1/2}, v_{k+1/2})}{\rho_k(h_k, v_k)} = \sum_{i=1}^d \log \frac{\psi(v_k^i)}{\psi(v_{k+1/2}^i)}, \quad (13)$$

with v^i denoting the i -th component of v . Indeed, we see that for each component of \tilde{v}_t

$$\psi'(\tilde{v}_t^i) = \frac{\partial_t(\psi(\tilde{v}_t^i))}{\partial_t \tilde{v}_t^i} = -\frac{\partial_t(\psi(\tilde{v}_t^i))}{\psi(\tilde{v}_t^i)} = -\partial_t(\log \psi(\tilde{v}_t^i)),$$

and so

$$\int_0^\eta \psi'(\tilde{v}_t^i) dt = \log \frac{\psi(\tilde{v}_0^i)}{\psi(\tilde{v}_\eta^i)} = \log \frac{\psi(v_k^i)}{\psi(v_{k+1/2}^i)},$$

from which (13) follows.

After this dissipative step, we apply a symplectic Hamiltonian integrator, such as

$$h_{k+1} = h_{k+1/2} + \eta v_{k+1}; \quad v_{k+1} = v_{k+1/2} - \eta \nabla_h \mathcal{C}_s(h_k).$$

This step conserves the density:

$$\rho_{k+1}(h_{k+1}, v_{k+1}) = \rho_{k+1/2}(h_{k+1/2}, v_{k+1/2}).$$

Indeed, we are applying to (h_k, v_k) the transformation

$$\begin{pmatrix} h \\ v \end{pmatrix} \mapsto \begin{pmatrix} h + \eta v - \eta^2 \nabla \mathcal{C}_s(h) \\ v - \eta \nabla \mathcal{C}_s(h) \end{pmatrix},$$

whose Jacobian $\begin{pmatrix} 1 - \eta^2 \Delta \mathcal{C}_s(h) & \eta \\ -\eta \Delta \mathcal{C}_s(h) & 1 \end{pmatrix}$ has determinant 1 for all h and v .

Following the above dynamics for K steps and applying the usual Markov argument we obtain the following result.

Proposition 12 *Consider the damped Hamiltonian dynamics described above, with \mathcal{C}_s twice differentiable on the whole \mathcal{H} . Let $\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a measurable function. Fix $K \in \mathbb{N}$ and let $\delta \in (0, 1)$. With probability at least $1 - \delta$ on the random draw $(s, h_0, v_0) \sim \mu^m \otimes \rho_0$, we have*

$$\Psi(\mathcal{L}_Z(h_K), \mathcal{L}_s(h_K)) \leq \log \frac{\xi}{\delta} + \log \frac{\rho_0(h_0, v_0)}{\rho_0(h_K, v_K)} + \sum_{k=0}^{K-1} \sum_{i=1}^d \log \frac{\psi(v_k^i)}{\psi(v_{k+1/2}^i)}, \quad (14)$$

with $\xi = \int_{\mathcal{Z}^m \times \mathcal{H}^2} e^{\Psi(\mathcal{L}_{\bar{s}}(h), \mathcal{L}_Z(h))} d\mu^m(\bar{s}) d\rho_0(h, v)$.

As a concrete example, we can choose $\psi(v) = \varepsilon|v|^p v$, for $p \geq 0$ and $\varepsilon > 0$, where $|\cdot|$ denotes the absolute value computed component-wise. The case $p = 0$ corresponds to the standard conformal damped Hamiltonian dynamics (França et al., 2020), and yields to

$$h_{k+1} = h_k + \eta v_{k+1}; \quad v_{k+1} = e^{-\varepsilon\eta} v_k - \eta \nabla_h \mathcal{C}_s(h_k),$$

with a density that increases exponentially as

$$\log \frac{\rho_{k+1}(h_{k+1}, v_{k+1})}{\rho_k(h_k, v_k)} = d\varepsilon\eta.$$

Note that this last term goes linearly with the dimension of the hypothesis space, a behaviour that is likely to bring poor bounds in over-parameterised settings. To avoid this, one can choose $p > 0$ and get

$$h_{k+1} = h_k + \eta v_{k+1}; \quad v_{k+1} = \frac{v_k}{(1 + p\varepsilon\eta|v_k|^p)^{1/p}} - \eta \nabla_h \mathcal{C}_s(h_k),$$

and

$$\log \frac{\rho_{k+1}(h_{k+1}, v_{k+1})}{\log \rho_k(h_k, v_k)} = \left(1 + \frac{1}{p}\right) \sum_{i=1}^d \log (1 + p\varepsilon\eta|v_k^i|^p).$$

With this choice, if the components of v are smaller than 1 (*e.g.*, when they are sampled from a Gaussian with small variance) the last term in the RHS of (14) will likely have a better behaviour with d . However, this improvement might come at the price of a larger $\log \frac{\rho_0(h_0, v_0)}{\rho_0(h_K, v_K)}$, due to the fact that less dissipative dynamics allow the model to explore a wider region of the hypothesis space, potentially ending up with a final state (h_K, v_K) with a ρ_0 density much lower than $\rho_0(h_0, v_0)$. It is unclear so far what is the optimal choice of ψ that can lead to tightest bounds. We leave the investigation of this open problem as future work.

5.2 Statement of authorship

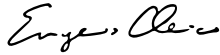
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).


Title of Paper	Generalisation under gradient descent via deterministic PAC-Bayes
Publication Status	Submitted
Publication Details	E. Clerico, T. Farghly, G. Deligiannidis, B. Guedj, and A. Doucet. Generalisation under gradient descent via deterministic PAC-Bayes. 2023.

Student Confirmation

Student Name:	Eugenio Clerico		
Contribution to the Paper	Tyler Farghly and I equally contributed to the paper. I came up with the initial idea in the continuous GD setting and proved the main results for continuous time. Tyler came up with the idea of extending the results to the discretised setting under smoothness assumptions. We worked together on the proofs in this setting, and on the applications and discussion of the results. George Deligiannidis, Benjamin Guedj, and Arnaud Doucet provided helpful insights and contributed to the writing of the paper and to the checking the proofs.		
Signature		Date	24/03/2023

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:	Prof George Deligiannidis		
Supervisor comments	Eugenio's description of his contributions to the paper is fair and accurate		
Signature		Date	27/03/2023

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 6

Discussion

This thesis explores several mathematical machine learning theory topics, leveraging the Gaussian limit of infinite-width networks and information-theoretic approaches to study generalisation and expressiveness for over-parameterised models.

Regarding expressiveness, Chapter 2 discusses the universality of networks with finite and infinite depth in the infinite-width limit, showing how the introduction of scaling factors allows for stable and expressive deep residual architectures. Adding to the theoretical analysis reported in this thesis, the empirical evidence in [Hayou et al. \(2021\)](#) highlights the potential practical value of the stable ResNets we introduced, which are shown to outperform their unscaled counterparts in several image recognition tasks.

Generalisation was the main focus of all the other chapters. Some of the results that we presented were merely theoretical, as is the case for the abstract framework underlying the duality between chained and unchained bounds (Chapter 4) and of the disintegrated PAC-Bayesian bounds that can apply to deterministic algorithms (Chapter 5). On the other hand, the discussion on the Gaussian PAC-Bayesian training (Chapter 3) is a clear example of how theoretical results can lead to the development of learning algorithms: the Gaussian limit that we establish for the infinitely wide shallow stochastic network not only has a direct application in the PAC-Bayesian method of Section 3.1, but has also later inspired the training algorithm of Section 3.2, which applies to a much broader range of architectures and achieved state-of-the-art PAC-Bayesian bounds.

In the rest of this section, we outline some of the open questions and potential avenues for further research that the analysis in this thesis has uncovered.

6.1 Limitations and open questions

6.1.1 Stable and expressive ResNets

The expressiveness analysis of Chapter 2 focuses on ReLU networks. However, we believe that most results would apply to more general settings. For instance, for “regular enough” non-polynomial activation functions, one would expect that the kernels Q_l become universal after a finite number of layers. Leveraging results from Daniely et al. (2016) and the proof techniques that we introduced in Section 2.6, one can find a power series for C_l of the form $\sum_{n=0}^{\infty} \alpha_n (x \cdot x')^n$. For a suitable class of activation functions, we conjecture that after a few layers the coefficients α will become all strictly positive, a sufficient condition for the universality of the kernel (Schoenberg, 1942). With a similar approach (maybe mimicking the proof of Proposition 2), we think it would be possible to get analogous results on any compact $K \subset \mathbb{R}^p$ (at least for $\sigma_b > 0$) and we conjecture that introducing depth- and layer-dependent scaling factors would allow for a stable and fully expressive infinite-depth limit for ResNets.

Another direction for future work is to look at expressiveness from a different perspective. So far, we have called expressive a process whose support is dense in L^2 . When the NTK governs the training dynamics, our definition is enough to ensure that any L^2 -function can be approximated with arbitrary precision. However, as extensively discussed in Yang and Salman (2019), the values of the eigenvalues in the Karhunen-Loève expansion of the Gaussian process play a crucial role in defining the performance of an algorithm. Indeed, having only a few eigenvalues consistently larger than the others leads to a process whose samples essentially lie in some low-dimensional space, potentially requiring a very long time to achieve good agreement with the data. Conversely, if there are too many dominant eigenvalues of the same order, the network has a tendency to overfit, as the implicit regularisation is extremely weak. It would be interesting to study the spectrum of the kernels and its dependence on the activation function ϕ , which could also contribute to the analysis of the activation function’s impact on the learning process. Moreover, examining the decay of the eigenvalues could provide interesting insights about the RKHS of the kernels. Recent research in this area includes Bietti and Bach (2021), which studies the asymptotic behaviour of some network-induced kernels on the sphere.

Finally, another interesting question is what happens to the expressiveness when the architecture’s width is finite and far from the kernel regime. Several works have studied the infinite-depth limit for finite-width architectures (*e.g.*, [Peluchetti and Favaro, 2020](#); [Li et al., 2022](#); [Hayou, 2023](#)), showing that there is not a universal law for the output (contrary to the Gaussian case). However, when the output is non-Gaussian, it is unclear if looking at the RKHS of the covariance kernels would yield any meaningful information about the expressive power of the model.

6.1.2 Gaussian PAC-Bayes

In [Section 3.1](#), we establish a Gaussian limit for the output of an infinitely wide stochastic network. The result is based on a central limit theorem that relies on the independence of the hidden nodes, a property that is lost for architectures with more than one hidden layer. A natural question is whether it is possible to describe the limit output distribution for multi-layer stochastic models. A few empirical tests suggest that, at the initialisation, the output remains Gaussian, consistently with the fact that the non-diagonal elements of all the covariance matrices tend to zero as the width grows. However, it is not yet clear if there is a learning regime where the correlation between hidden nodes stays weak enough to ensure the Gaussianity of the output throughout the network’s training. In any case, even if a Gaussian limit could be established, obtaining a practical learning algorithm (as the one that we propose for the shallow case) would require finding a suitable compressed approximation of the covariance matrices of the hidden layers, as their large size ($n \times n$ for a network of width n) would be a severe bottleneck for the algorithm’s implementation.

Another interesting question is whether it is possible to describe the learning dynamics of an infinitely wide stochastic network via the neural tangent kernel. The standard derivation of the NTK evolution ([Jacot et al., 2018](#)) relies on the fact that the optimisation objective \mathcal{C}_s depends on the parameters h only through the network’s output F , making it possible to expand $\nabla \mathcal{C}_s = \nabla F \cdot \nabla_F \mathcal{C}_s$. Nevertheless, when the learning objective is a PAC-Bayesian bound, it contains a relative entropy term that depends directly on the trainable hyper-parameters. We are currently working on establishing alternative NTK dynamics for a “regularised” learning. It turns out that if the objective is in the form $\mathcal{C}_s = \mathcal{L}_s + \frac{\lambda}{2} \|h - \hat{h}\|^2$ (where \hat{h} is the value of

the parameters at the initialisation), then the evolution is governed by

$$\partial_t F_i(x) = -\lambda(F_i(x) - \hat{F}_i(x)) - \frac{1}{m} \sum_{z' \in \mathcal{S}} \sum_j \theta_{ij}(x, x') \partial_{F_j} \ell(x', w),$$

where \hat{F} is the network's output at the initialisation. Under suitable assumptions, this result can be used to study the dynamics of a shallow wide stochastic network from an NTK perspective. We remark that [Huang et al. \(2022\)](#) also tackled the NTK evolution for the PAC-Bayesian training of a shallow stochastic network, but used a different approach that heavily relies on the specific setting considered (*i.e.*, real output, quadratic loss, and training of the hidden layer only).

6.1.3 Chained generalisation bounds

The general framework of Chapter 4 encompasses several information-theoretic results from the literature and establishes how each can be associated with a chained bound. However, from a practical perspective, it is not yet clear how powerful the results brought by this framework can be. For instance, computing these information-theoretic bounds is not feasible in most problems of interest: they require evaluating the expectation under the unknown training data set distribution $\mathbb{P}_{\mathcal{S}}$. Yet, upper bounds on the mutual information have allowed for empirical bounds for some stochastic iterative optimisation algorithms ([Bu et al., 2019](#); [Haghifam et al., 2020](#); [Rodríguez-Gálvez et al., 2020](#); [Neu et al., 2021](#)). However, the chaining technique has not yet been exploited in this context and might lead to interesting results.

Recently, [Haghifam et al. \(2023\)](#) demonstrated that several variants of the mutual information bound from [Russo and Zou \(2019\)](#) cannot achieve min-max rates in the context of stochastic convex optimisation. This raises concerns about whether these results are the right ones to pursue in order to gain a better understanding of generalisation for over-parameterised models. In any case, they did not consider the chaining technique or bounds based on information-theoretic measures other than mutual information. Future research may extend their analysis to these settings as well.

Finally, Chapter 4 is mainly focused on the backwards-channel (from the data set to the hypothesis) information-theoretic perspective, which seemingly pairs more naturally with the chaining on the hypotheses' space. However, the chained PAC-Bayesian result that we

present is an example of a forwards-channel bound, as it considers the distribution of the hypotheses conditioned on the sample. A future direction of study could be to extend our general framework to include forwards-channel bounds. We believe this should not present significant technical difficulties and might bring new valuable insights.

6.1.4 Deterministic PAC-Bayes under gradient descent

The final chapter of this thesis (Chapter 5) presents novel bounds that are completely computable but have not yet been adequately tested empirically. Future research will involve conducting experiments on benchmark learning tasks to draw a deeper comparison between our results and existing literature. However, the smoothness condition, which is necessary for our bound to hold, may be difficult to check for general settings. Thus, further investigation is needed to identify more easily verifiable assumptions.

Another direction for research would involve focusing on the infinite width limit, where the network’s output behaves as a Gaussian process at the initialisation and is described by the neural tangent kernel dynamics during the training. In this setting, the results from [Jacot et al. \(2020\)](#), characterising the network’s Laplacian, could contribute to the analysis of our bounds. Moreover, one could adopt a functional perspective by examining how the output density evolves. While PAC-Bayesian bounds in expectation for Gaussian processes exist (*e.g.*, [Seeger, 2002](#)), we are not aware of any disintegrated versions of them. For this approach, one major challenge is the need for well-defined densities of distributions on functional spaces, which often requires a high degree of technical sophistication.

Lastly, it should be noted that the bounds proposed in Chapter 5 explode as the training time approaches infinity. One potential solution to this problem is introducing noise to the training dynamics (*e.g.*, stochastic gradient Langevin dynamics), which would cause the output density to converge towards the Gibbs posterior. However, this would also make evaluating the final density for a finite time horizon T much more challenging and may render the bound not explicitly computable. An alternative approach to achieve finite generalisation bounds would be to impose additional regularity assumptions on the optimisation objective. For example, one could use our results to obtain a finite bound for a time T' and then leverage other techniques to analyse how the population loss for $h_{T'}$ differs from that of h_T , for any arbitrarily large time horizon T .

Bibliography

- P. Alquier. PAC-Bayesian bounds for randomized empirical risk minimizers. *Mathematical Methods of Statistics*, 17, 2008.
- P. Alquier. User-friendly introduction to PAC-Bayes bounds. *Preprint arXiv:2110.11216*, 2021.
- P. Alquier and G. Biau. Sparse single-index model. *Journal of Machine Learning Research*, 14, 2013.
- P. Alquier and K. Lounici. PAC-Bayesian bounds for sparse regression estimation with exponential weights. *Electronic Journal of Statistics*, 5, 2010.
- P. Alquier, J. Ridgway, and N. Chopin. On the properties of variational approximations of Gibbs posteriors. *Journal of Machine Learning Research*, 17, 2016.
- J.M. Antognini. Finite size corrections for neural network Gaussian processes. *ICML Workshop*, 2019.
- S. Arora, S. Du, W. Hu, Z. Li, R. Salakhutdinov, and R. Wang. On exact computation with an infinitely wide neural net. *NeurIPS*, 2019.
- A.R. Asadi and E. Abbe. Chaining meets chain rule: Multilevel entropic regularization and training of neural nets. *Journal of Machine Learning Research*, 21, 2020.
- A.R. Asadi, E. Abbe, and S. Verdú. Chaining mutual information and tightening generalization bounds. *NeurIPS*, 2018.
- J.Y. Audibert. Aggregated estimators and empirical complexity for least square regression. *Annales de l'Institut Henri Poincaré : Probabilités et statistiques*, 40(6), 2004.
- J.Y. Audibert and O. Bousquet. PAC-Bayesian generic chaining. *NeurIPS*, 2004.

- J.Y. Audibert and O. Catoni. Robust linear least squares regression. *The Annals of Statistics*, 39(5), 2011.
- P.L. Bartlett, Y. Freund, W.S. Lee, and R.E. Schapire. Boosting the margin: a new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5), 1998.
- P.L. Bartlett, O. Bousquet, and S. Mendelson. Local Rademacher complexities. *The Annals of Statistics*, 33(4), 2005.
- P.L. Bartlett, D.J. Foster, and M.J. Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.
- R. Bassily, S. Moran, I. Nachum, J. Shafer, and A. Yehudayoff. Learners that use little information. *ALT*, 2018.
- A. Basteri and D. Trevisan. Quantitative Gaussian approximation of randomly initialized deep neural networks. *Preprint arXiv:2203.07379*, 2022.
- A. Bietti and F. Bach. Deep equals shallow for relu networks in kernel regimes. *ICLR*, 2021.
- F. Biggs and B. Guedj. Differentiable PAC-Bayes objectives with partially aggregated neural networks. *Entropy*, 23(10), 2021.
- F. Biggs and B. Guedj. On margins and derandomisation in PAC-Bayes. *AISTATS*, 2022.
- F. Biggs, V. Zantedeschi, and B. Guedj. On margins and generalization for voting classifiers. *NeurIPS*, 2022.
- S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities - A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.
- O. Bousquet, S. Boucheron, and G. Lugosi. *Introduction to Statistical Learning Theory*. Springer, 2004.
- O. Bousquet, Y. Klochkov, and N. Zhivotovskiy. Sharper bounds for uniformly stable algorithms. *COLT*, 2020.
- D. Bracale, S. Favaro, S. Fortini, and S. Peluchetti. Large-width functional asymptotics for deep Gaussian neural networks. *ICLR*, 2021.

- Y. Bu, S. Zou, and V. V. Veeravalli. Tightening mutual information based bounds on generalization error. *ISIT*, 2019.
- S. Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4), 2015.
- O. Catoni. A PAC-Bayesian approach to adaptive classification. *preprint LPMA 840*, 2003.
- O. Catoni. *Statistical Learning Theory and Stochastic Optimization*. Ecole d'Eté de Probabilités de Saint-Flour. Springer, 2004.
- O. Catoni. PAC-Bayesian supervised classification: The thermodynamics of statistical learning. *IMS Lecture Notes Monograph Series*, 2007.
- Z. Charles and D. Papailiopoulos. Stability and generalization of learning algorithms that converge to global optima. *ICML*, 2018.
- B.E. Chérif-Abdellatif. Convergence rates of variational inference in sparse deep learning. *ICML*, 2020.
- E. Clerico, G. Deligiannidis, and A. Doucet. Conditionally Gaussian PAC-Bayes. *AISTATS*, 2022a.
- E. Clerico, A. Shidani, G. Deligiannidis, and A. Doucet. Chained generalisation bounds. *COLT*, 2022b.
- E. Clerico, G. Deligiannidis, and A. Doucet. Wide stochastic networks: Gaussian limit and PAC-Bayesian training. *ALT*, 2023a.
- E. Clerico, T. Farghly, G. Deligiannidis, B. Guedj, and A. Doucet. Generalisation under gradient descent via deterministic PAC-Bayes. *Preprint arXiv:2209.02525*, 2023b.
- N. Cohen, O. Sharir, and A. Shashua. On the expressive power of deep learning: A tensor analysis. *COLT*, 2016.
- C. Cortes and V.N. Vapnik. Support-vector networks. *Machine Learning*, 20(3), 1995.
- G.V. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2, 1989.

- A.S. Dalalyan and A. Tsybakov. Aggregation by exponential weighting, sharp PAC-Bayesian bounds and sparsity. *Machine Learning*, 72(1-2), 2008.
- A. Daniely, R. Frostig, and Y. Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. *NeurIPS*, 2016.
- S. De and S.L. Smith. Batch normalization biases residual blocks towards the identity function in deep networks. *NeurIPS*, 2020.
- L. Devroye and T.J. Wagner. Distribution-free performance bounds for potential function rules. *IEEE Transactions on Information Theory*, 25, 1979.
- R.M. Dudley. The sizes of compact subsets of Hilbert space and continuity of Gaussian processes. *Journal of Functional Analysis*, 1(3), 1967.
- R.M. Dudley. *Real Analysis and Probability*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2 edition, 2002.
- G.K. Dziugaite and D.M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *UAI*, 2017.
- G.K. Dziugaite and D.M. Roy. Data-dependent PAC-Bayes priors via differential privacy. *NeurIPS*, 31, 2018.
- G.K. Dziugaite, K. Hsu, W. Gharbieh, G. Arpino, and D.M. Roy. On the role of data in PAC-Bayes bounds. *AISTATS*, 2021.
- R. Eldan and O. Shamir. The power of depth for feedforward neural networks. *COLT*, 2016.
- A. Elisseeff, T. Evgeniou, and M. Pontil. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6(3), 2005.
- A. R. Esposito, M. Gastpar, and I. Issa. Generalization error bounds via Rényi-, f -divergences and maximal leakage. *IEEE Transactions on Information Theory*, 67(3), 2021.
- V. Feldman and J. Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. *COLT*, 2019.

- K.I. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3), 1989.
- A. Garriga-Alonso, C.E. Rasmussen, and L. Aitchison. Deep convolutional networks as shallow gaussian processes. *ICLR*, 2019.
- P. Germain, A. Lacasse, F. Laviolette, and M. Marchand. Pac-bayesian learning of linear classifiers. *ICML*, 2009.
- P. Germain, F. Bach, A. Lacoste, and S. Lacoste-Julien. PAC-Bayesian theory meets Bayesian inference. *NeurIPS*, 2016.
- I. Goodfellow, A. Courville, and Y. Bengio. *Deep learning*. MIT Press, 2016.
- P. Grunwald, T. Steinke, and L. Zakyntinou. PAC-Bayes, MAC-Bayes and conditional mutual information: Fast rate bounds that handle general VC classes. *COLT*, 2021.
- B. Guedj. A primer on PAC-Bayesian learning. *Second congress of the French Mathematical Society*, 2019.
- B. Guedj and P. Alquier. PAC-Bayesian estimation and prediction in sparse additive models. *Electronic Journal of Statistics*, 7, 2013.
- M. Haddouche, B. Guedj, O. Rivasplata, and J. Shawe-Taylor. PAC-Bayes unleashed: Generalisation bounds with unbounded losses. *Entropy*, 23(10), 2021.
- M. Haghifam, J. Negrea, A. Khisti, D.M. Roy, and G.K. Dziugaite. Sharpened generalization bounds based on conditional mutual information and an application to noisy, iterative algorithms. *NeurIPS*, 2020.
- M. Haghifam, G.K. Dziugaite, S. Moran, and D.M. Roy. Towards a unified information-theoretic framework for generalization. *NeurIPS*, 2021.
- M. Haghifam, B. Rodríguez-Gálvez, R. Thobaben, M. Skoglund, D.M. Roy, and K.G. Dziugaite. Limitations of information-theoretic generalization bounds for gradient descent methods in stochastic convex optimization. *ALT*, 2023.
- B. Hanin. Random neural networks in the infinite width limit as Gaussian processes. *Preprint arXiv:2107.01562*, 2021.

- B. Hanin and M. Sellke. Approximating continuous functions by ReLU nets of minimal width. *Preprint arXiv:1710.11278*, 2018.
- M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. *ICML*, 2016.
- S. Hayou. On the infinite-depth limit of finite-width neural networks. *Preprint arXiv:2210.00688*, 2023.
- S. Hayou, A. Doucet, and J. Rousseau. Mean-field behaviour of neural tangent kernel for deep neural networks. *Preprint arXiv:1905.13654*, 2019a.
- S. Hayou, A. Doucet, and J. Rousseau. On the impact of the activation function on deep neural networks training. *ICML*, 2019b.
- S. Hayou, E. Clerico, B. He, G. Deligiannidis, A. Doucet, and J. Rousseau. Stable ResNet. *AISTATS*, 2021.
- K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 1989.
- W. Hu, L. Xiao, and J. Pennington. Provable benefit of orthogonal initialization in optimizing deep linear networks. *ICLR*, 2020.
- W. Huang, C. Liu, Y. Chen, T. Liu, and Richard Y. Da X. Demystify optimization and generalization of over-parameterized PAC-Bayesian learning. *Preprint arXiv:2202.01958*, 2022.
- A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: convergence and generalization in neural networks. *NeurIPS*, 2018.
- A. Jacot, F. Gabriel, and C. Hongler. The asymptotic spectrum of the Hessian of DNN throughout training. *ICML*, 2020.
- M. Kanagawa, P. Hennig, D. Sejdinovic, and B. Sriperumbudur. Gaussian processes and kernel methods: A review on connections and equivalences. *Preprint arXiv:1807.02582*, 2018.
- P. Kidger and T. Lyons. Universal approximation with deep narrow networks. *COLT*, 2020.

- O. Kounchev. *Multivariate Polysplines: Applications to Numerical and Wavelet Analysis*. Elsevier Science, 2001.
- S. Lang. *Real and Functional Analysis*. Graduate Texts in Mathematics. Springer, 3rd edition, 2012.
- J. Langford and R. Caruana. (Not) bounding the true error. *NeurIPS*, 2002.
- J. Langford and M. Seeger. Bounds for averaging classifiers. *CMU tech report*, 2001.
- J. Langford and J. Shawe-Taylor. Pac-bayes & margins. *NeurIPS*, 2002.
- Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521, 2015.
- J. Lee, Y. Bahri, R. Novak, S. Schoenholz, J. Pennington, and J. Sohl-Dickstein. Deep neural networks as Gaussian processes. *ICLR*, 2018.
- J. Lee, L. Xiao, S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. *NeurIPS*, 2019.
- J. Lee, S. Schoenholz, J. Pennington, B. Adlam, L. Xiao, R. Novak, and J. Sohl-Dickstein. Finite versus infinite neural networks: an empirical study. *NeurIPS*, 2020.
- M. Li, M. Nica, and D. Roy. The neural covariance SDE: Shaped infinite depth-and-width networks at initialization. *NeurIPS*, 2022.
- A.T. Lopez and V. Jog. Generalization error bounds using Wasserstein distances. *IEEE Information Theory Workshop (ITW)*, 2018.
- A. Lovelace. Notes upon L.F. Menabrea’s “Sketch of the analytical engine invented by Charles Babbage”, 1842.
- Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang. The expressive power of neural networks: A view from the width. *NeurIPS*, 2017.
- G. Lugosi and G. Neu. Generalization bounds via convex analysis. *COLT*, 2022.
- A. G. Matthews, J. Hron, M. Rowland, R.E. Turner, and Z. Ghahramani. Gaussian process behaviour in wide deep neural networks. *ICLR*, 2018.

- A. Maurer. A note on the PAC Bayesian theorem. *Preprint arXiv:0411099*, 2004.
- D.A. McAllester. Some PAC-Bayesian theorems. *COLT*, 1998.
- D.A. McAllester. PAC-Bayesian model averaging. *COLT*, 1999.
- D.A. McAllester. PAC-Bayesian stochastic model selection. *Machine Learning*, 51, 2003a.
- D.A. McAllester. Simplified PAC-Bayesian margin bounds. *COLT*, 2003b.
- Z. Mhammedi, P. Grünwald, and B. Guedj. PAC-Bayes un-expected Bernstein inequality. *NeurIPS*, 2019.
- C. Micchelli, Y. Xu, and H. Zhang. Universal kernels. *Journal of Machine Learning Research*, 7, 2006.
- G.F. Montufar, R. Pascanu, K. Cho, and Y. Bengio. On the number of linear regions of deep neural networks. *NeurIPS*, 2014.
- R.M. Neal. Bayesian learning for neural networks. *Springer Science & Business Media*, 118, 1995.
- G. Neu, G.K. Dziugaite, M. Haghifam, and D.M. Roy. Information-theoretic generalization bounds for stochastic gradient descent. *COLT*, 2021.
- B. Neyshabur, S. Bhojanapalli, and N. Srebro. A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. *ICLR*, 2018.
- R. Novak, L. Xiao, J. Lee, Y. Bahri, G. Yang, J. Hron, D. A Abolafia, J. Pennington, and J. Sohl-Dickstein. Bayesian deep convolutional networks with many channels are gaussian processes. *ICLR*, 2019.
- A.B. Novikoff. On convergence proofs on perceptrons. *Symposium on the Mathematical Theory of Automata*, 1962.
- E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor, and S. Sun. PAC-Bayes bounds with data dependent priors. *Journal of Machine Learning Research*, 13(112), 2012.
- V. I. Paulsen and M. Raghupathi. *An introduction to the theory of reproducing kernel Hilbert spaces*. Cambridge University Press, 2016.

- S. Peluchetti and S. Favaro. Infinitely deep neural networks as diffusion processes. *AISTATS*, 2020.
- B. Poole, S. Lahiri, M. Raghu, J. Sohl-Dickstein, and S. Ganguli. Exponential expressivity in deep neural networks through transient chaos. *NeurIPS*, 2016.
- M. Pérez-Ortiz, O. Risvapata, J. Shawe-Taylor, and C. Szepesvári. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22, 2021a.
- M. Pérez-Ortiz, O. Rivasplata, B. Guedj, M. Gleeson, J. Zhang, J. Shawe-Taylor, M. Bober, and J. Kittler. Learning PAC-Bayes priors for probabilistic neural networks. *Preprint arXiv:2109.10304*, 2021b.
- O. Rivasplata, I. Kuzborskij, C. Szepesvári, and J. Shawe-Taylor. PAC-Bayes analysis beyond the usual bounds. *NeurIPS*, 2020.
- B. Rodríguez-Gálvez, G. Bassi, R. Thobaben, and M. Skoglund. On random subset generalization error bounds and the stochastic gradient Langevin dynamics algorithm. *IEEE Information Theory Workshop (ITW)*, 2020.
- D. Rolnick and M. Tegmark. The power of deeper networks for expressing natural functions. *ICML*, 2018.
- D. Russo and J. Zou. How much does your data exploration overfit? Controlling bias via information usage. *IEEE Transactions on Information Theory*, 66(1), 2019.
- I. Safran and O. Shamir. Depth-width tradeoffs in approximating natural functions with neural networks. *ICML*, 2017.
- I.J. Schoenberg. Positive definite functions on spheres. *Duke Mathematical Journal*, 9, 1942.
- S. Schoenholz, J. Gilmer, S. Ganguli, and J. Sohl-Dickstein. Deep information propagation. *ICLR*, 2017.
- M. Seeger. PAC-Bayesian Generalization Error Bounds for Gaussian Process Classification. *Journal of Machine Learning Research*, 3, 2002.
- S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.

- J. Shawe-Taylor and R.C. Williamson. A PAC analysis of a Bayesian estimator. *COLT*, 1997.
- B. Sriperumbudur, K. Fukumizu, and G. Lanckriet. Universality, characteristic kernels and RKHS embedding of measures. *Journal of Machine Learning Research*, 12(70), 2011.
- T. Steinke and L. Zakyntinou. Reasoning about generalization via conditional mutual information. *COLT*, 2020.
- I. Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2, 2001.
- I. Steinwart. Convergence types and rates in generic Karhunen-Loève expansions with applications to sample path properties. *Potential Analysis*, 51, 2019.
- M. Talagrand. Majorizing measures: the generic chaining. *The Annals of Probability*, 24(3), 1996.
- M. Telgarsky. Representation benefits of deep feedforward networks. *Preprint arXiv:1509.08101*, 2015.
- M. Telgarsky. Benefits of depth in neural networks. *COLT*, 2016.
- I. O Tolstikhin and Y. Seldin. PAC-Bayes-empirical-Bernstein inequality. *NeurIPS*, 2013.
- V.N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, 2000.
- H. Wang, M. Diaz, J. C. S. Santos Filho, and F. P. Calmon. An information-theoretic view of generalization via Wasserstein distance. *ISIT*, 2019.
- A. Xu and M. Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. *NeurIPS*, 2017.
- G. Yang. Tensor programs I: Wide feedforward or recurrent neural networks of any architecture are Gaussian processes. *NeurIPS*, 2019a.
- G. Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *Preprint arXiv:1902.04760*, 2019b.

- G. Yang. Tensor programs II: Neural tangent kernel for any architecture. *Preprint arXiv:2006.14548*, 2020a.
- G. Yang. Tensor programs III: Neural matrix laws. *Preprint arXiv:2009.10685*, 2020b.
- G. Yang and E.J. Hu. Tensor programs IV: Feature learning in infinite-width neural networks. *ICML*, 2021.
- G. Yang and E. Littwin. Tensor programs IIb: Architectural universality of neural tangent kernel training dynamics. *ICML*, 2021.
- G. Yang and H. Salman. A fine-grained spectral perspective on neural networks. *Preprint arXiv:1907.10599*, 2019.
- G. Yang, E.J. Hu, I. Babuschkin, S. Sidor, X. Liu, D. Farhi, N. Ryder, J. Pachocki, W. Chen, and J. Gao. Tensor programs V: Tuning large neural networks via zero-shot hyperparameter transfer. *NeurIPS*, 2021.
- C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *ICLR*, 2017.
- C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3), 2021.
- H. Zhang, Y. N. Dauphin, and T. Ma. Fixup initialization: Residual learning without normalization. *ICLR*, 2019.
- W. Zhou, V. Veitch, M. Austern, R.P. Adams, and P. Orbanz. Non-vacuous generalization bounds at the ImageNet scale: a PAC-Bayesian compression approach. *ICLR*, 2018.