The twin peaks of learning neural networks

Elizaveta Demyanenko[1], Christoph Feinauer[1], Enrico M. Malatesta[1], Luca Saglietti[1]

[1] *Department of Computing Sciences, Bocconi University, 20136 Milano, Italy*

Contents

Abstract

Recent works demonstrated the existence of a double-descent phenomenon for the generalization error of neural networks, where highly overparameterized models escape overfitting and achieve good test performance, at odds with the standard bias-variance trade-off described by statistical learning theory. In the present work, we explore a link between this phenomenon and the increase of complexity and sensitivity of the function represented by neural networks. In particular, we study the Boolean mean dimension (BMD), a metric developed in the context of Boolean function analysis. Focusing on a simple teacher-student setting for the random feature model, we derive a theoretical analysis based on the replica method that yields an interpretable expression for the BMD, in the high dimensional regime where the number of data points, the number of features, and the input size grow to infinity. We find that, as the degree of overparameterization of the network is increased, the BMD reaches an evident peak at the interpolation threshold, in correspondence with the generalization error peak, and then slowly approaches a low asymptotic value. The same phenomenology is then traced in numerical experiments with different model classes and training setups. Moreover, we find empirically that adversarially initialized models tend to show higher BMD values, and that models that are more robust to adversarial attacks exhibit a lower BMD.

1 Introduction

The evergrowing scale of modern neural networks often prevents a detailed understanding of how predictions relate back to the model inputs [Sejnowski, 2020]. While this lack of interpretability can hinder adoption in sectors with a high impact on society [Rudin, 2019], the impressive performance of neural network-based models in fields like natural language processing [Vaswani et al., 2017, OpenAI, 2023, Touvron et al., 2023], computational biology [Jumper et al., 2021] and computer vision and image generation [Ramesh et al., 2022, Rombach et al., 2022] have made them the de-facto standard for many real-world applications. This tension has motivated a large interest in the field of explainable AI (XAI) [Montavon et al., 2018, Guidotti et al., 2018, Vilone and Longo, 2020].

Deep learning models, which by now can feature hundreds of billions of parameters [Brown et al., 2020], seemingly defy the notion that increasing model complexity should decrease generalization performance. Counter to what one would expect from statistical learning theory [Vapnik, 1999], the observation has been that larger —heavily overparameterized— models often perform better [Neyshabur et al., 2017]. This has led to the question how complex the function represented by an overparameterized neural network is after training. Many lines of research suggest that neural network models are biased towards implementing simple functions, despite their large parameter count, and that this implicit bias is crucial for their good generalization performance [Valle-Perez et al., 2018]. The general problem of measuring the complexity of deep neural networks has given rise to several complexity metrics [Novak et al., 2018] and studies on how they relate to generalization [Jiang et al., 2019].

Connected to this, recent studies [Geiger et al., 2019, Belkin et al., 2019] on the effect of overparameterization in neural networks led to the rediscovery of the "double descent" phenomenon, first observed in the statistical physics literature [Opper, 1995], which is the observation that when increasing the capacity of a neural network (measured, for example, by the number of parameters) the generalization error shows a sudden peak around the interpolation point (where approximately zero training error is achieved), but then a second decrease towards a low asymptotic value is observed at higher overparameterization.

In the present work, we study the double descent phenomenon under a notion of function sensitivity based on the *mean dimension* [Hahn et al., 2022, Hoyt and Owen, 2021]. The mean dimension yields a measure of the mean interaction order between input variables in a function, and can also be proved to be related to the variance of the function under local perturbations of the input features. While this notion originated in the field statistics [Liu and Owen, 2006], several computational techniques have been proposed for its estimation in the context of neural

networks. One of the main obstacles, however, comes from trying to characterize the sensitivity of the function over an input distribution that is strongly structured and not fully known.

In this paper, we propose to focus on the study of the *Boolean mean dimension* [O'Donnell, 2014] (BMD), which involves a simple i.i.d. binary input distribution. We show how the BMD can be estimated efficiently, and provide analytical and numerical evidence of the correlation of this metric with several phenomena observed on the data used for training and testing the model.

2 Related works

2.1 Overparameterization and Double Descent

Several studies [Baity-Jesi et al., 2018, Geiger et al., 2019, Advani et al., 2020] confirmed the robustness of the double descent phenomenology for a large variety of architectures, datasets, and learning paradigms. An analytical study of double descent in the context of the random feature model [Rahimi et al., 2007] was conducted rigorously for the square loss in [Mei and Montanari, 2019] and for generic loss by [Gerace et al., 2020] using the replica method [Mézard et al., 1987]. Double descent has then later found also in the context of one layer model learning a Gaussian mixture dataset [Mignacco et al., 2020]; similarly to the random feature model, the peak in the generalization can be avoided by optimally regularizing the network. In this context in [Baldassi et al., 2020] it was also shown that choosing the optimal regularization corresponds to maximize a flatness-based measure of the loss minimizer. A range of later studies further explored this phenomenology in related settings [d'Ascoli et al., 2020, Gerace et al., 2022].

Different scenarios have also been shown to give rise to a similar phenomenology, such as the epoch-wise double descent and sample non-monotonicity [Nakkiran et al., 2021] and the triple descent that can appear with noisy labels and can be regularized by the non-linearity of the activation function [d'Ascoli et al., 2020].

In this work, we connect the usual double descent of the generalization error with the behavior of the mean dimension, which is a complexity metric that can be evaluated without requiring task-specific data.

2.2 Mean Dimension and Boolean Mean Dimension

The *mean dimension* (MD), based on the analysis of variance (ANOVA) expansion [Efron and Stein, 1981, Owen, 2003], can be intuitively understood as a marker of the complexity of a function due to the presence of interactions between a large set of input variables.

The mean dimension has been used as a tool to analyze and compare for example neural networks [Hoyt and Owen, 2021, Hahn et al., 2022] and, with a slightly different definition, also generative models of protein sequences [Feinauer and Borgonovo, 2022]. The MD has the advantage that it can be calculated for a *black-box function*, without regard to the internal mechanism for calculating the input-output relation. One major drawback, however, is the intense computational cost associated with its direct estimation. This computational limitation has led to the proposal of several approximation strategies [Hoyt and Owen, 2021, Hahn et al., 2022]. In some special cases, the mean dimension can be explicitly expressed as a function of the coefficients of a Fourier expansion, as seen from the relationship between the Boolean Mean Dimension (BMD) and the *total influence* [O'Donnell, 2014] defined in the analysis of Boolean functions (see below), and its generalization [Feinauer and Borgonovo, 2022] for functions with categorical variables.

3 Mean Dimension

In the next paragraphs, we first provide a general mathematical definition of the mean dimension for a square-integrable function with real-valued input distribution. We then specialize to the case of a binary input distribution and define the Boolean Mean Dimension (BMD), which will be the main quantity investigated throughout this paper. Finally, we will discuss how to efficiently estimate the MD and the BMD through a simple Monte Carlo procedure.

3.1 Mathematical Definition

To give a proper mathematical definition of the mean dimension, for a real-valued function $f(\boldsymbol{x})$ of n variables $f : \mathbb{R}^n \to \mathbb{R}$, it is convenient to introduce some notation that will be used in the rest of the paper. We will denote the set of indexes $\{1, \dots n\}$ by $[n]$. We define $\boldsymbol{x}_u$ the set of input variables x_i, with $i \in u \subseteq [n]$ and by $\boldsymbol{x}_{\setminus u}$ the set of variables for which $i \notin u$. We will also assume that $\boldsymbol{x}$ is drawn from a distribution $p(\boldsymbol{x})$. The basic idea of the mean dimension [Hahn et al., 2022] is to derive a complexity measure for f from an expansion of the type

$$f(\boldsymbol{x}) = \sum_{u \subseteq [n]} f_u(\boldsymbol{x}_u) \tag{1}$$

where the "components" $f_u(\boldsymbol{x}_u)$ can be computed from the following recursion relation

$$f_u(\boldsymbol{x}_u) \equiv \int f(\boldsymbol{x})\, p(\boldsymbol{x}_{\setminus u}|\boldsymbol{x}_u)\, d\boldsymbol{x}_{\setminus u} - \sum_{v \subset u} f_v(\boldsymbol{x}_v) \tag{2}$$

with the initial condition $f_\emptyset = \int f(\boldsymbol{x})\, p(\boldsymbol{x})\, d\boldsymbol{x} \equiv \mathbb{E}[f]$. It can be shown that coefficients of the expansion have zero average if u is non empty

$$\int f_u(\boldsymbol{x}_u) p_u(\boldsymbol{x}_u)\, d\boldsymbol{x}_u = 0 \qquad u \neq \emptyset \tag{3}$$

where we have denoted by $p_u(\boldsymbol{x}_u)$ the marginal probability distribution over the set u. Moreover, they satisfy orthogonality relations, namely

$$\int f_u(\boldsymbol{x}_u) f_v(\boldsymbol{x}_v) p_{u\cup v}(\boldsymbol{x}_{u\cup v})\, d\boldsymbol{x}_{u\cup v} = 0\,, \qquad \text{if } u \neq v\,. \tag{4}$$

Using those relations we can write the variance of the function as a decomposition of $2^n - 1$ terms

$$\sigma^2 = \mathbb{E}[f^2] - \mathbb{E}[f]^2 = \sum_{u \subseteq [n] \setminus \emptyset} \sigma_u^2 \tag{5}$$

where

$$\sigma_u^2 \equiv \int f_u^2(\boldsymbol{x}_u)\, p_u(\boldsymbol{x}_u)\, d\boldsymbol{x}_u\,. \tag{6}$$

The mean dimension M_f is then defined as [Hahn et al., 2022]

$$M_f = \sum_{u \subseteq [n]} |u| \frac{\sigma_u^2}{\sigma^2}, \tag{7}$$

i.e. a weighted sum over possible interactions, with each subset of inputs contributing based on how much they influence the variance.

3.2 Pseudo-Boolean Functions and Fourier coefficients

We now derive an explicit expression for the mean dimension of n-dimensional pseudo-Boolean functions taking values on the real domain, $f : \{-1, 1\}^n \to \mathbb{R}$ under the assumption of input features that are i.i.d. from $\{-1, 1\}$.

Denoting by $\boldsymbol{s} \in \{-1, 1\}^n$ the n-dimensional binary input of f, such a function can be uniquely written as a *Fourier expansion* [O'Donnell, 2014] in terms of a finite set of *Fourier coefficients* $\hat{f}_u$, $u \subseteq [n]$ as

$$f(\boldsymbol{s}) = C + \sum_i h_i s_i + \sum_{i<j} J_{ij} s_i s_j + \sum_{i<j<k} K_{ijk} s_i s_j s_k + \ldots = \sum_{u \subseteq [n]} \hat{f}_u \, \chi_u(\boldsymbol{s}_u) \tag{8}$$

where

$$\chi_u(\boldsymbol{s}_u) = \prod_{i \in u} s_i \tag{9}$$

represent the Fourier basis of the decomposition that are orthonormal $\langle \chi_u(\boldsymbol{s}) \chi_v(\boldsymbol{s}) \rangle = \delta_{u,v}$ with respect to the uniform distribution over $\{-1, 1\}^n$, where we use the notation

$$\langle \bullet \rangle \equiv \frac{1}{2^n} \sum_{\boldsymbol{s} \in \{-1,1\}^n} \bullet \, . \tag{10}$$

The Fourier coefficients $\hat{f}_u$ can give information about the moments of the function f with respect to the uniform distribution (10) over $\boldsymbol{s}$; for example the first moment is

$$\langle f(\boldsymbol{s}) \rangle = \hat{f}_\emptyset \tag{11}$$

whereas the variance can be obtained as

$$\sigma^2 = \langle f^2(\boldsymbol{s}) \rangle - \langle f(\boldsymbol{s}) \rangle^2 = \sum_{u \subseteq [n] \setminus \emptyset} \hat{f}_u^2 \, . \tag{12}$$

We can quantify the contribution c_k of interaction of order k to the variance of $f(\boldsymbol{s})$ as the ratio

$$c_k = \frac{\sum_{u \subseteq [n] \setminus \emptyset : |u| = k} \hat{f}_u^2}{\sigma^2} \, . \tag{13}$$

Notice that $\sum_k c_k = 1$, so that c_k can be interpreted as a (discrete) probability measure over interactions. The mean dimension of f can then be written as the mean interaction degree when weighted according to it contribution to the variance, i.e. as a weighted sum of feature influences divided by the total variance of the function, so

$$M_f \equiv \sum_{k=1}^{n} k c_k = \frac{\sum_{u \subseteq [n]} |u| \, \hat{f}_u^2}{\sigma^2} \tag{14}$$

This expression is equivalent to Eq. (7) for pseudo-Boolean functions under the assumptions that all features are i.i.d from $\{-1, 1\}$. The expression connects the notion of simplicity in terms of variance contributions to the same notion in terms of explicit expansion coefficients. Intuitively, a large mean dimension is indicating that the function fluctuates due to a large contribution of high-degree interactions.

3.3 Estimating the Mean Dimension through Monte Carlo

The expression of the mean dimension in (7) involves a sum over all the set of subsets of n variables, and its numerical evaluation through a brute-force approach would be intractable in high dimension. However, it can be shown that a more efficient evaluation scheme of equation (7),

can be achieved through a Monte Carlo approach [Liu and Owen, 2006]. First, the MD can be rewritten as a sum over the n input components:

$$M_f = \frac{\sum_{i=1}^{n} \tau_i^2}{\sigma^2} \tag{15}$$

where the *influence* of the i-th input component τ_i is defined as:

$$\tau_i^2 = \frac{1}{2} \int d\boldsymbol{x}\, dx_i'\, p(\boldsymbol{x})\, p(x_i'|\boldsymbol{x}_{\setminus i})(f(\boldsymbol{x}) - f(\boldsymbol{x}^{\oplus i}))^2\,. \tag{16}$$

and where we have denoted by $\boldsymbol{x}^{\oplus i}$ a vector $\boldsymbol{x}$ with a resampled i_{th} coordinate. We show an original proof of this identity in Appendix A.

Note that the definition of the MD for a generic input distribution in Eq. (16), entails a resampling procedure that presumes knowledge of the conditional distribution of a pixel given the rest of the pixel values. In the general case, this pixel is to be resampled multiple times from this conditional distribution, to compute the variance of the function under this variation of the input. This conditional distribution, however, is not a known quantity for a real dataset. For this reason, for example, some authors have proposed an "exchange" procedure, where one randomply samples a different pixel value observed in the same dataset [Hahn et al., 2022], however this approximation neglects the within sample correlations.

Expression (16) can be specialized to the case of binary i.i.d. inputs, where one can identify the influence functions τ_i^2 with the discrete derivatives:

$$\tau_i^2 = \left\langle (\mathcal{D}_i f(\boldsymbol{s}))^2 \right\rangle \tag{17}$$

where $\mathcal{D}_i f(\boldsymbol{s})$ denotes the i_{th} (discrete) derivative of $f(\boldsymbol{s})$, i.e.

$$\mathcal{D}_i f(\boldsymbol{s}) \equiv \frac{f(s_1, \ldots, s_i = 1, \ldots, s_n) - f(s_1, \ldots, s_i = -1, \ldots, s_n)}{2} \tag{18}$$

and measures the average sensitivity of the function to a flip of the i_{th} variable. The sum of the influences $\sum_i \left\langle (\mathcal{D}_i f(\boldsymbol{s}))^2 \right\rangle$ is known in the field of the analysis of pseudo-Boolean functions as *total influence* of f [O'Donnell, 2014]. In terms of the Fourier expansion, we have

$$\mathcal{D}_i f(\boldsymbol{s}) = \sum_{u \subseteq [n]: i \in u} \hat{f}_u\, \chi_{u\setminus i}(\boldsymbol{s}_{u\setminus i}) \tag{19}$$

Therefore computing the mean dimension for pseudo-Boolean functions boils down to querying the function f on uniformly sampled binary sequences of length $n-1$.

3.4 Boolean Mean Dimension

In the general case, the underlying input distribution of the training dataset is not known and estimating the MD on this distribution becomes unfeasible. In the present work, we propose employing the estimation procedure presented in the last section, based on binary sequences, as an easily computable proxy of the sensitivity of the neural network function. In order to distinguish this proxy from the mean dimension over the dataset distribution, we call the resulting quantity the Boolean Mean Dimension (BMD). We show in the results below that the BMD can in some cases be computed analytically, and that it is qualitatively related to the generalization phenomenology in neural networks.

4 Analytical results

We now derive an analytic expression for the mean dimension in the special case of the random feature model [Rahimi et al., 2007, Goldt et al., 2019, Loureiro et al., 2021, Baldassi et al., 2022], focusing on the same high dimensional regime where the double descent phenomenon can be detected. In the next sections, we will define the model, the learning task and the high dimensional limit precisely, and we will sketch the analytical derivation of the expression for the Boolean Mean Dimension.

4.1 Model definition and learning task

The random feature model (RFM) is a two-layer neural network with random and fixed first-layer weights (also called features) and trainable second-layer weights. Given a D-dimensional input, $\boldsymbol{x} \in \mathbb{R}^D$, and denoting by $F \in \mathbb{R}^{D\times N}$ the $D \times N$ frozen feature matrix, the pre-activation of the RFM is given by:

$$\hat{y}(\boldsymbol{w}; \boldsymbol{x}) = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \, \sigma \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k \right) \tag{20}$$

where $\boldsymbol{w}$ is an N-dimensional weight vector and σ is a (usually non-linear) function. The parameter N indicates the number of features in the RFM and can be varied to change the degree of over-parametrization of the model. As in [Baldassi et al., 2022], we will hereafter focus on the case of i.i.d. standard normal distributed feature components $F_{ki} \sim \mathcal{N}(0, 1)$, although the formalism allows for a simple extension to a generic fixed feature map, under a simple weak correlation requirement (see [Gerace et al., 2020, Loureiro et al., 2021] for additional details).

We consider a classification task defined by a training dataset of size P, denoted as $\mathcal{D} = \{\boldsymbol{x}^\mu, y^\mu\}_{\mu=1}^P$. The inputs are assumed to be i.i.d. with first and second moments fixed respectively to $\mathbb{E}x_i = 0$ and $\mathbb{E}x_i^2 = 1$. Note that, for example, both binary input components $x_i \in \{-1, 1\}$ and Gaussian components $x_i \sim \mathcal{N}(0, 1)$ satisfy the above assumption. The binary labels $y^\mu \in \{-1, 1\}$ are assumed to be produced by a "teacher" linear model $\boldsymbol{w}^T \in \mathbb{R}^D$, with normalized weights on the D-sphere $\|\boldsymbol{w}^T\|_2^2 = D$, according to:

$$y^\mu = \text{sign} \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} w_k^T x_k^\mu \right) , \qquad \mu \in [P] \, . \tag{21}$$

The learning task is then framed as an optimization problem with generic loss function ℓ and ridge regularization

$$\boldsymbol{w}_\star = \underset{\boldsymbol{w}\in\mathbb{R}^N}{\arg\min} \left[\sum_{\mu=1}^{P} \ell(y^\mu, \, \hat{y}^\mu(\boldsymbol{w}; \boldsymbol{x}^\mu)) + \frac{\lambda}{2} \|\boldsymbol{w}\|_2^2 \right] , \tag{22}$$

where λ is a positive external parameter controlling the regularization strength. In the following we will consider the two most common convex loss functions, namely the mean squared error (MSE) and the cross-entropy (CE) losses, defined as

$$\ell_{mse}(y, \hat{y}) = \frac{1}{2} (y^\mu - \hat{y}^\mu)^2 \tag{23a}$$

$$\ell_{ce}(y, \hat{y}) = \log \left(1 + e^{-y\hat{y}} \right) . \tag{23b}$$

We analyze the learning problem in the high-dimensional limit where the number of features, input components and training-set size diverge $N \, , \, D \, , \, P \to \infty$ at constant rates $\alpha \equiv P/N = \mathcal{O}(1)$ and $\alpha_D \equiv D/N = \mathcal{O}(1)$. In this limit, strong concentration properties allow for a deterministic characterization of the above-defined learning problem in terms of a finite set of scalar quantities called order parameters. In the next sections, and in detail in the appendices, we will sketch the derivation of this reduced description.

4.2 Rephrasing the problem in terms of the Boltzmann measure

The learning task in (22) can be characterized within a statistical physics framework. One can introduce a probability measure over the weights $\boldsymbol{w}$ in terms of the Boltzmann distribution

$$\boldsymbol{w} \sim p_\beta(\boldsymbol{w}; \mathcal{D}) = \frac{e^{-\beta \sum_{\mu=1}^{P} \ell(y^\mu, \hat{y}^\mu(\boldsymbol{w}; \boldsymbol{x}^\mu)) - \frac{\beta\lambda}{2} \sum_{i=1}^{N} w_i^2}}{Z_\beta} \tag{24}$$

where β is the inverse temperature, the loss function in (22) plays the role of an energy, and the partition function Z_β is a normalization factor that reads

$$Z_\beta = \int d\boldsymbol{w}\, e^{-\beta \sum_{\mu=1}^{P} \ell(y^\mu, \hat{y}^\mu(\boldsymbol{w}; \boldsymbol{x}^\mu)) - \frac{\beta\lambda}{2} \sum_{i=1}^{N} w_i^2} \,. \tag{25}$$

The distribution $p_\beta(\boldsymbol{w}; \mathcal{D})$ can be interpreted in a Bayesian setting as the posterior distribution over the weights $\boldsymbol{w}$ given a dataset $\mathcal{D}$, and (24) corresponds to Bayes theorem, where the term $e^{-\beta \sum_\mu \ell(y^\mu, \hat{y}^\mu(\boldsymbol{w}; \boldsymbol{x}^\mu))}$, corresponds to the likelihood and $e^{-\frac{\beta\lambda}{2}\|\boldsymbol{w}\|_2^2}$ is the prior distribution over the weights.

In the zero-temperature limit, when $\beta \to \infty$, the probability measure $p_\beta(\boldsymbol{w}; \mathcal{D})$ concentrates on the solutions to the optimization problem in (22). To characterize the typical (i.e. the most probable) properties of these solutions, one needs to perform an average over the possible realizations of the training set $\mathcal{D}$ and of the features F, computing the free-energy of the system

$$f = -\lim_{\beta\to\infty} \lim_{N\to\infty} \frac{1}{\beta N} \mathbb{E}_{\mathcal{D},F} \ln Z_\beta \quad . \tag{26}$$

The computation of this "quenched" average can be achieved via the replica method [Mézard et al., 1987] from spin-glass theory, which reduces the characterization of the solutions of (22) to the determination of a finite set of scalar quantities called order parameters [Engel and Van den Broeck, 2001, Malatesta, 2023].

In appendix B.1, we sketch the replica calculation for the free energy, first presented in [Gerace et al., 2020], in the simplifying case of an odd non-linear activation σ.

4.3 Analytical determination of the BMD in the RFM

We now derive an analytic expression for the Boolean Mean Dimension (BMD) which can be efficiently evaluated for a trained RFM. The definition (15) reads

$$M_f(\boldsymbol{w}) \equiv \frac{\frac{1}{2}\sum_{k=1}^{D} \left\langle (\hat{y}(\boldsymbol{w}; \boldsymbol{x}) - \hat{y}(\boldsymbol{w}; \boldsymbol{x}^{\oplus k}))^2 \right\rangle}{\langle \hat{y}^2(\boldsymbol{w}; \boldsymbol{x}) \rangle - \langle \hat{y}(\boldsymbol{w}; \boldsymbol{x}) \rangle^2} \,. \tag{27}$$

where $\langle \bullet \rangle$ and $x^{\oplus k}$, defined in (10) and (16), entail an expectation over i.i.d. uniform binary inputs. In appendix B.2, we perform the annealed averages appearing in the numerator and the denominator separately, obtaining the expression:

$$M_f(\boldsymbol{w}) = \frac{\frac{1}{N}\sum_{ij} \bar{\Psi}_{ij} w_i w_j}{\frac{1}{N}\sum_{ij} \Psi_{ij} w_i w_j} \tag{28}$$

where we defined

$$\Omega_{ij} \equiv \frac{1}{D} \sum_{k=1}^{D} F_{ki} F_{kj} \,. \tag{29a}$$

$$\bar{\Psi}_{ij} \equiv \bar{\kappa}_\star^2\, \Omega_{ii} \mathbb{I}_{ij} + \bar{\kappa}_0^2\, \Omega_{ij} + \bar{\kappa}_1^2\, \Omega_{ij}^2 \,, \tag{29b}$$

$$\Psi_{ij} \equiv \kappa_\star^2\, \mathbb{I}_{ij} + \kappa_1^2\, \Omega_{ij} \,. \tag{29c}$$

and the coefficients κ are defined as expectations of derivatives of the activation function over a standard Gaussian measure $Dz = \frac{e^{-z^2/2}}{\sqrt{2\pi}} dz$:

$$\kappa_0 = \int Dz\, \sigma(z)\,, \qquad \kappa_1 = \int Dz\, \sigma'(z)\,, \qquad \kappa_2 = \int Dz\, \sigma^2(z)\,, \tag{30a}$$

$$\bar{\kappa}_0 = \kappa_1\,, \qquad \bar{\kappa}_1 = \int Dz\, \sigma''(z)\,, \qquad \bar{\kappa}_2 = \int Dz\, \left(\sigma'(z)\right)^2\,, \tag{30b}$$

$$\kappa_\star^2 = \kappa_2 - \kappa_1^2 - \kappa_0^2\,, \qquad \bar{\kappa}_\star^2 = \bar{\kappa}_2 - \bar{\kappa}_1^2 - \bar{\kappa}_0^2\,. \tag{30c}$$

As we show in the appendix, the above expression (28) is universal: evaluating the MD with respect to a different i.i.d. input distributions with matching first and second moments would give exactly the same result.

Moreover, note that the evaluation of expression (28) no longer involves a Monte-Carlo over the input distribution, with a major gain in computational cost. In appendix B.2, we show the agreement of this compact formula with the computationally more expensive Monte Carlo estimation of the BMD.

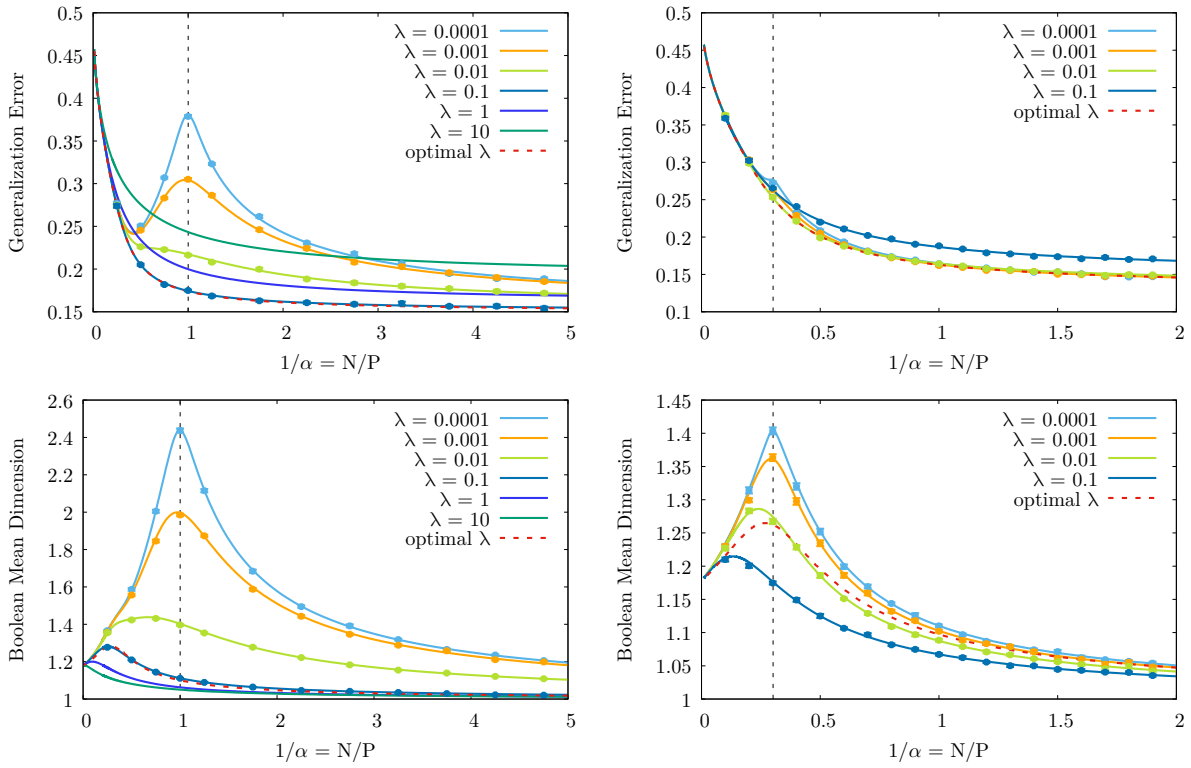

Figure 1: Generalization error (top panels) and BMD (bottom panels) as a function of the overparameterization degree $1/\alpha = N/P$, for fixed $\alpha_T = P/D = 3$ and with $\sigma = \tanh$. The left and right panels represents respectively the case of the MSE and of the CE loss. Several value of the regularization λ are displayed, together with the optimal one (which was found by minimizing the generalization error for each value of α, see red dashed line). As it can be seen in both plots, for small regularization λ, the location of the peak in the generalization error exactly coincides with the one in the BMD (vertical dashed lines). As one increases the regularization the peak in the both the generalization and the BMD is milded.

The mean dimension therefore explicitly depends on the model parameters $\boldsymbol{w}$. The evaluation of the typical BMD of a trained RFM can thus be computed by taking an expectation over the zero-temperature Boltzmann measure for the weights derived in the replica computation,

$M_f = \mathbb{E}_{\mathcal{D},F} \langle M_f(\boldsymbol{w}) \rangle_{\boldsymbol{w}}$. The notation $\langle \bullet \rangle_{\boldsymbol{w}} \equiv \int d\boldsymbol{w} \bullet p_\infty(\boldsymbol{w}; \mathcal{D})$ is thus used to indicate an average over the posterior distribution in equation (24), in the large β limit.

In the case of the replica computation for an odd activation function, that we reported in appendix B.1, one can simplify further expression (28) by recognizing that $\bar{\kappa}_1 = 0$ and that $\Omega_{ii} = 1$ when the feature components have second moment equal to 1. In this case, the numerator and the denominator can be directly expressed in terms of the order parameters of the model:

$$M_f = 1 + (\bar{\kappa}_2 - \kappa_2) \frac{q_d}{Q_d} \tag{31}$$

where

$$q_d \equiv \mathbb{E}_{\mathcal{D},F} \left\langle \frac{1}{N} \sum_{i=1}^{N} w_i^2 \right\rangle_{\boldsymbol{w}} , \tag{32a}$$

$$p_d \equiv \mathbb{E}_{\mathcal{D},F} \left\langle \frac{1}{N} \sum_{i,j=1}^{N} \Omega_{ij} w_i w_j \right\rangle_{\boldsymbol{w}} , \tag{32b}$$

$$Q_d \equiv \kappa_\star^2 q_d + \kappa_1^2 p_d \, . \tag{32c}$$

The order parameters q_d, p_d can be computed by solving saddle point equations as shown in Appendix B.1.

Notice that in the case of a linear activation function the BMD is always 1 since a flip in the inputs will induce always the same response.

In Fig. 1 we show the plot of the generalization error and the corresponding BMD of the RFM at a fixed α_T, as a function of $1/\alpha$ for the MSE (left panels) and CE loss (right panels). As shown in [Gerace et al., 2020], for small regularization λ the generalization error develops a peak approximately where the model starts to fit all training data. In the case of the MSE loss, this threshold is often called interpolation threshold and it is located at $N = P$. When using the CE loss, this happens when the projected data become linearly separable and the exact location of the threshold strongly depends on the input statistics and features. Exactly in the correspondence of the generalization error peak the BMD displays its own peak, meaning that the function implemented by the network is more sensitive to perturbation of the inputs.

An interesting insight can be deduced from the behavior of the BMD at the optimal value of regularization for the RFM (dashed red curves in Fig. 1). While the generalization error becomes monotonic as the over-parametrization is increased, the BMD still reaches a peak at first and then descends to 1 only in the kernel limit $N/P \to \infty$. This might be surprising since the ground-truth linear model, the teacher, has BMD equal to 1 and one would expect the best generalizing RFM to achieve the best possible approximation of this function and therefore to match its BMD. However, blind minimization of the BMD is not compatible with good generalization, as seen from the performance of the RFM with very large regularization λ. The explanation of this comes from the architectural mismatch between the linear teacher and the RFM: according to the GET the RFM learning problem is equivalent to a linear problem with an additional noise with an intensity regulated by the degree of non-linearity of the activation function [d'Ascoli et al., 2020]. This noise initially forces the under-parameterized RFM to overstretch its parameters to fit the data, causing an increased sensitivity to input perturbations. As the over-parameterization is increased, the RFM becomes equivalent to an optimally regularized linear model [Gerace et al., 2020] and the BMD slowly drops to 1 in this limit.

Note that in the large dataset limit, when $\alpha, \alpha_T \to \infty$ with $\alpha_D = \mathcal{O}(1)$, a secondary peak for the BMD of the RFM emerges around $\alpha_D = 1$, i.e. when the number of parameters of the RFM is the same as the number of input features. This peak is caused by the insurgence of singular values in the spectrum of the covariance matrix Ω and is more accentuated at lower values of the regularization. Since modern deep networks operate in a completely different regime from the large dataset limit specified above, we expect this secondary peak not to be visible in

realistic settings. For example, in the above plots in the low regularization regime, this peak is overshadowed by the main BMD peak. We analyze this phenomenology in detail in appendix C.

5 Numerical results

In the following subsections, we explore numerically the robustness of the BMD phenomenology analyzed in the RFM, considering different types of data distribution, model architecture and learning task.

Furthermore, we show that adversarially initialized models also display higher BMD, and that the increased sensitivity associated with a large BMD can hinder the robustness of the model against random perturbations of the training inputs.

Finally, we show that the location of the BMD peak is robust to the choice of input statistics used for its measurement, even in non-i.i.d. settings.

5.1 Experimental setup

In the following subsections, each panel displays the performance of a large number of different model architectures with varying degree of over-parameterization, trained on different datasets. Except where specified otherwise, all model are initialized with the common *Xavier* method [Glorot and Bengio, 2010] and use the Adam optimizer [Kingma and Ba, 2014], with batch size 128 and learning rate 10^{-4}. No specific early stopping criterion is implemented. As in other works analysing the double descent, we experiment with different levels of uniformly random label noise during training (which is introduced by corrupting a random fraction of labels), which tends to make the double descent peak more pronounced [Nakkiran et al., 2021]. We discuss the effect of label noise below.

5.1.1 Model architectures

We consider different types of model architectures:

- Random feature model (RFM), described above, where the number of hidden neurons in the first (fixed) layer controls the degree of over-parameterization.
- Two-layer fully-connected network (MLP) with tanh activation, where the number of hidden neurons in the first layer controls the degree of over-parameterization.
- ResNet-18: a family of minimal ResNet [He et al., 2016] architectures based on the implementation of [Nakkiran et al., 2021]. The structure is finalized with fully connected and softmax layers. As in [Nakkiran et al., 2021], we control the over-parameterization of the model by changing the number of channels in the convolutional layers. Namely, the 4 ResNet blocks contain convolutional layers of widths $[k, 2k, 4k, 8k]$, with k varying from 1 to 20.

Both RFM and two-layer fully connected networks in our experiments use hyperbolic tangent activation functions and have weights initialized from a Gaussian distribution and bias terms initialized with zeros. The loss function optimized during training is the cross-entropy loss with L_2 regularization (the intensity of the regularization is set to zero if not specified otherwise).

5.1.2 Data preprocessing

In the following experiments, we use continuous inputs during the training of the models, normalizing the input features to lie within the $[-1, 1]$ interval. While such normalizations are common in preprocessing pipelines, here this procedure has also the benefit of matching the range of variability of the training inputs with that of the randomly i.i.d. sampled binary

sequences used to estimate the BMD. We explore the effect of different normalization ranges in Appendix Sec. E.

5.2 MD and generalization peaks as a function of overparametrization

In Fig. 2 we show train and test error, and the BMD for an RFM trained with and without label noise on binary MNIST (even vs odd digits) as a function of the hidden layer width. In Fig. 3, we instead consider a two-layer MLP trained on 10-digits MNIST (varying width) and a ResNet-18 trained on CIFAR10 (varying number of channels), both with label noise. In the multi-label case, we are defining the BMD of the network as the average of the BMDs over the classes, where the output of the network is a vector of predicted log-probabilities for each class (i.e. there is a log-softmax activation in the last layer).

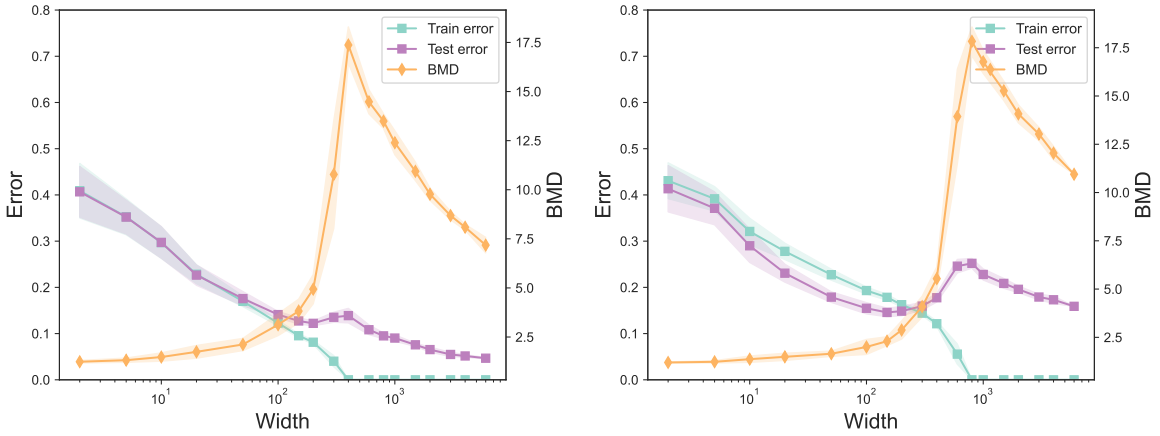

Figure 2: Train error (turquoise), test error (violet) and BMD (orange) curves of the random feature model trained on the MNIST dataset with binary labels on 5K train samples with 0% label noise (left) and 10% label noise (right), tested on 5K samples. The resulting plots represent an average (and standard deviations) obtained repeating 20 different times the experiment.

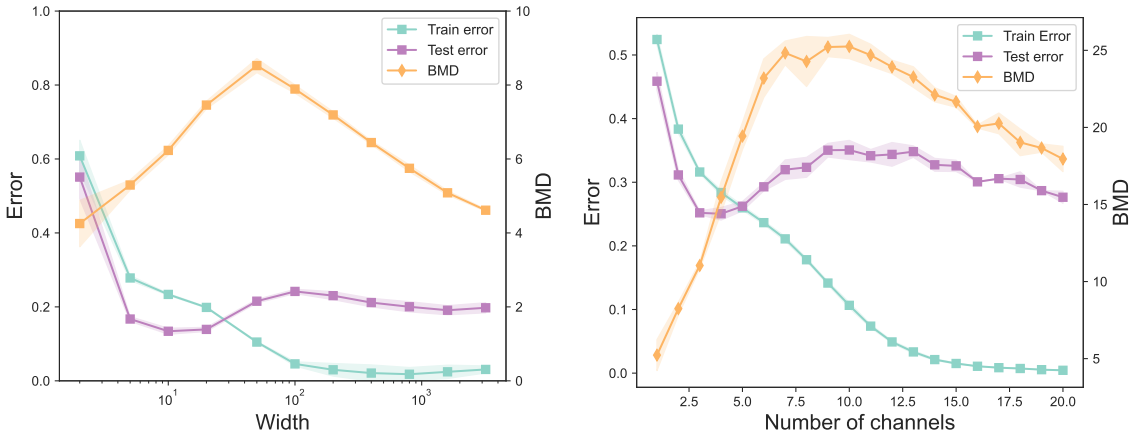

Figure 3: (Left) Train error (turquoise), test error (violet) and BMD (orange) of the two-layer fully-connected network trained on the MNIST dataset with 10 labels on 20K train samples with 20% label noise, tested on the 5K samples. The resulting plot represents an average (and standard deviations) obtained repeating 20 different times the experiment. (Right) Train error, test error and BMD of ResNet-18 trained on the CIFAR-10 with 15% label noise in the train set. The resulting plot represents an average (and standard deviations) obtained repeating 5 different times the experiment.

Position of the BMD peak The BMD displays a peak around the point where the number of parameters of the model allows it to reach zero training error, in close correspondence with the generalization error peak. We find this phenomenology to be robust with respect to the model class, the dataset, and the over-parameterization procedure. Notice however, that standard optimizers based on SGD are able to implicitly regularize the trained models and can strongly reduce the peaking behavior, as already observed in the context of double descent. In the presented figures we introduced label-noise, which ensures the presence of over-fitting and is thus able to restore both peaks.

An important observation is that, in order to see this phenomenology, it is not necessary to account for the training input distribution for the evaluation of the MD, which would not be possible in the case of real data. In fact, in the over-fitting regime, it is possible to detect an increased sensitivity of the neural network function for multiple input distributions, including the i.i.d. binary inputs entailed in the BMD evaluation. This is explored further in sub-section 5.6.

Asymptotic behavior of the BMD When the degree of parametrization of the model is further increased, the BMD decreases and settles on an asymptotic value. The decrease of the BMD in the number of parameters is faster with lower label noise, see Fig. 2 (left panel vs. right panel). The asymptotic value, reached in the limit of an infinite number of parameters, is task- and model-dependent. For example, in Fig. 2, the functions learned by the RFMs no longer approximate a linear model (BMD equal to 1), and are instead bound to higher values of the BMD.

Visibility of the BMD peak and Label Noise The double-descent generalization peak can be a very subtle phenomenon when the learning task is too coherent and the noise level in the data is too weak. With this type of data, the phenomenon can be made more evident [Nakkiran et al., 2021] by adding label noise to the training data. This strategy naturally reduces the signal-to-noise ratio and increases the over-fitting potential during training. The BMD peak, however, seems to be easily identifiable even with zero label noise, (see left panel of Fig. 2) where the generalization peak is less pronounced. Note that the BMD does not require any data (neither training nor test) in order to be estimated, so it can be used as a black-box test for assessing the proximity to the separability threshold and therefore as a signal of over-fitting.

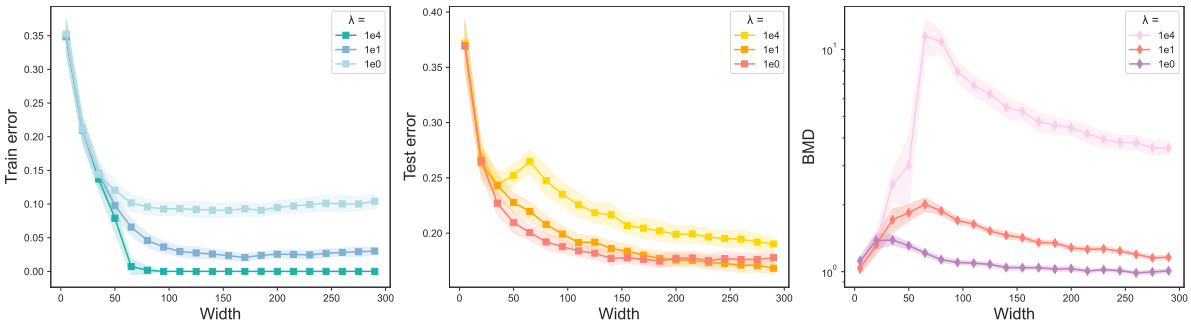

Figure 4: Impact of training with L2 regularization of the random feature model using the MNIST dataset with 10 labels, 200 train samples with no label noise and evaluating on 5K test samples. The loss used is cross-entropy. In all plots, the curves are colored by the strength of the regularization weight λ. (Left) Regularization effect on the train error. (Center) Regularization effect on the test error. (Right) Regularization effect on the BMD. The generalization error smoothly decreases with the degree of over-parameterization. Similarly, the BMD peak can be dampened by adding stronger regularization.

Impact of regularization It has been shown that regularizing the model weakens the double-descent peak and that, at the optimal value of the regularization intensity, the generalization error smoothly decreases with the degree of over-parameterization. Similarly, the BMD peak can be dampened by adding stronger regularization, as shown in Fig. 4.

5.2.1 BMD and Training Set Size

In this section, we investigate the effect of varying the number of training samples for a fixed model capacity and training procedure. By increasing the number of training samples, starting from a low number, the same model can switch from being over- to under-parameterized. Therefore increasing the number of training samples has two effects on the test error curve: on the one hand, increasing the number of training samples decreases the test error, shifting the test error curve mostly downwards. On the other hand, increasing the number of training samples increases the capacity at which the double descent peak occurs since a higher capacity is needed until the training set is effectively memorized. This shifts the test error curve (and the BMD curve) to the right. This effect can be seen in Figure 5.

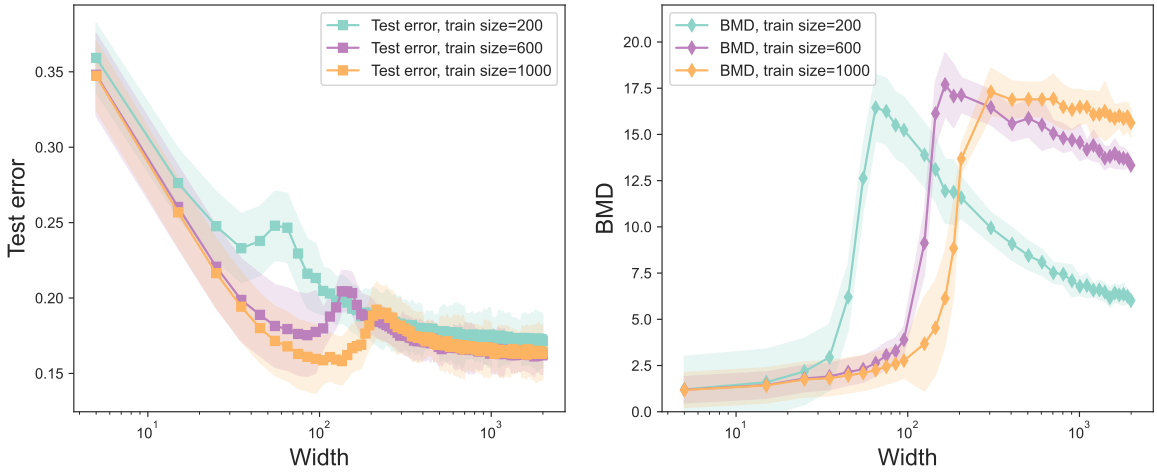

Figure 5: Effect of changing the training set size on the test error (left panel) and the BMD (right panel) of the random feature model using the MNIST dataset with 10 labels, with no label noise and evaluating on 200 test samples. The resulting plot represents an average (and standard deviations) obtained repeating 15 different times the experiment.

5.3 BMD and Adversarial Initialization

In this section, we analyze the BMD of two-layer fully connected networks under adversarial initialization [Liu et al., 2020] on the MNIST dataset. This initialization scheme can be used to artificially hinder the generalization performance of the model, forcing it to converge on a bad minimum of the loss. We here aim to show that the initialization has also an effect on the BMD of the model, increasing the sensitivity of the network.

The adversarial initialization protocol works as follows. We train a two-layer fully connected network in two different phases: in the first phase, we push the network towards an adversarial initialization by pretraining the model with 100% label noise for a fixed amount of epochs; in the second phase, we train the model on the original dataset, with no label noise, for 200 epochs. The resulting plot, in Fig. 6 (left panel), represents an average over 15 different realizations of the experiment and shows the effect of the length of the pretraining phase on both generalization

performance and BMD of the network. In agreement with our analysis, we observe a simultaneous increase of the two metrics when the adversarial initialization phase is longer and the network is driven towards worse generalization.

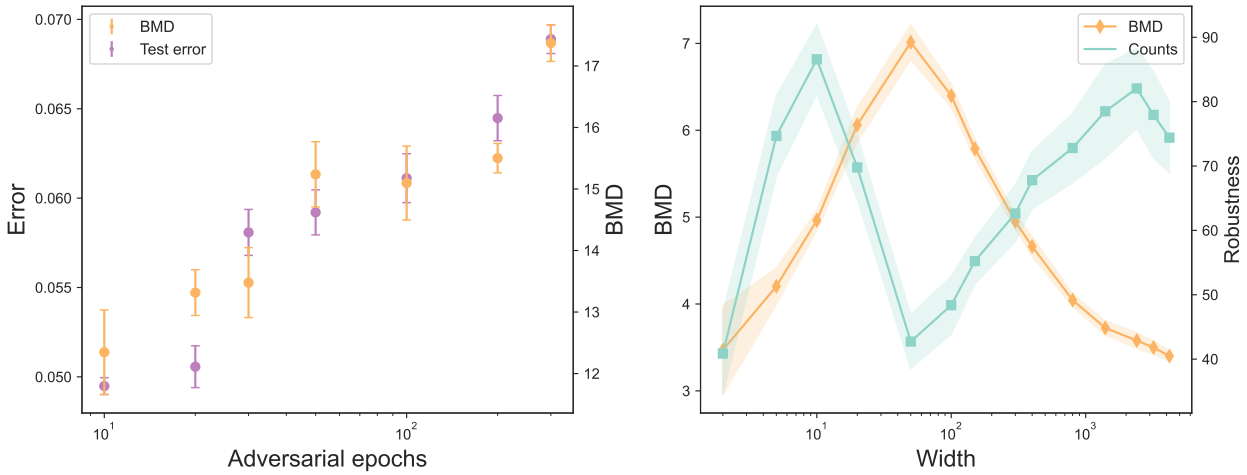

Figure 6: (Left): BMD (orange points), and test error (violet points) estimated for a two-layer fully connected network of width 10^3, trained according to the adversarial initialization protocol described in section 5.3 on 20K samples of the MNIST dataset. On the horizontal axis we vary the number of pretraining epochs, and plot the corresponding increase in the generalization error and the BMD of the model after the second learning stage. The points represent an average of 15 different realizations of the experiment. (Right): BMD (orange line) and counts (turquoise line) estimated for a two-layer fully connected network trained on the MNIST dataset using 20K train samples with 20% label noise and tested on 5K samples. Counts represent the average amount of sign flips of random pixels of a correctly predicted test image that are necessary to fool the model to a wrong class label. The amount is averaged over all the correctly predicted test data samples. The resulting plot represents an average of 40 different realizations of the experiment. We observe that higher values of the BMD correspond to lower robustness of the model and vice versa.

5.4 BMD and Robustness Against Adversarial Attacks

In this section, we analyze the connection between BMD of a model and its robustness to adversarial attacks. We consider a two-layer fully-connected network trained on MNIST with 10 classes. We define as our robustness measure the average count of sign flips of randomly chosen pixels, needed to change the model prediction on a test sample that was previously classified correctly. The lower the counts, the lower the robustness of the model. Varying the capacity of the model by varying the width of the hidden layer, we plot this robustness measure against the BMD of the model in Fig. 6 (right panel). We observe that BMD and robustness strongly anti-correlate, with the peak in BMD coinciding with a minimum of robustness.

5.5 Pixel-Wise Contributions to BMD

The MD as expressed in eq (15) is proportional to a sum of contributions τ_i^2 of single features indexed by i. Similar to [Hahn et al., 2022], we plot these contributions in Fig. 7 as a heatmap, where the bright spots indicate features that contribute strongly to the MD. We show four

heatmaps, corresponding to different capacities and at different distances from the BMD peak, for a two-layer fully connected network trained on MNIST.

Note that the colors are normalized to the $[0, 1]$ range, so that very bright spots correspond to pixels that contribute to the BMD the most. It can be seen that for under-parametrized networks few pixels give the largest contribution to the BMD. Near the BMD peak, a large fraction of the pixels in the center of the image dominate the BMD, and for even larger capacities we again have fewer pixels with maximal values. This can be interpreted as the classifier losing "focus" at the interpolation point and paying attention to fewer patterns in the over-parametrized regime.

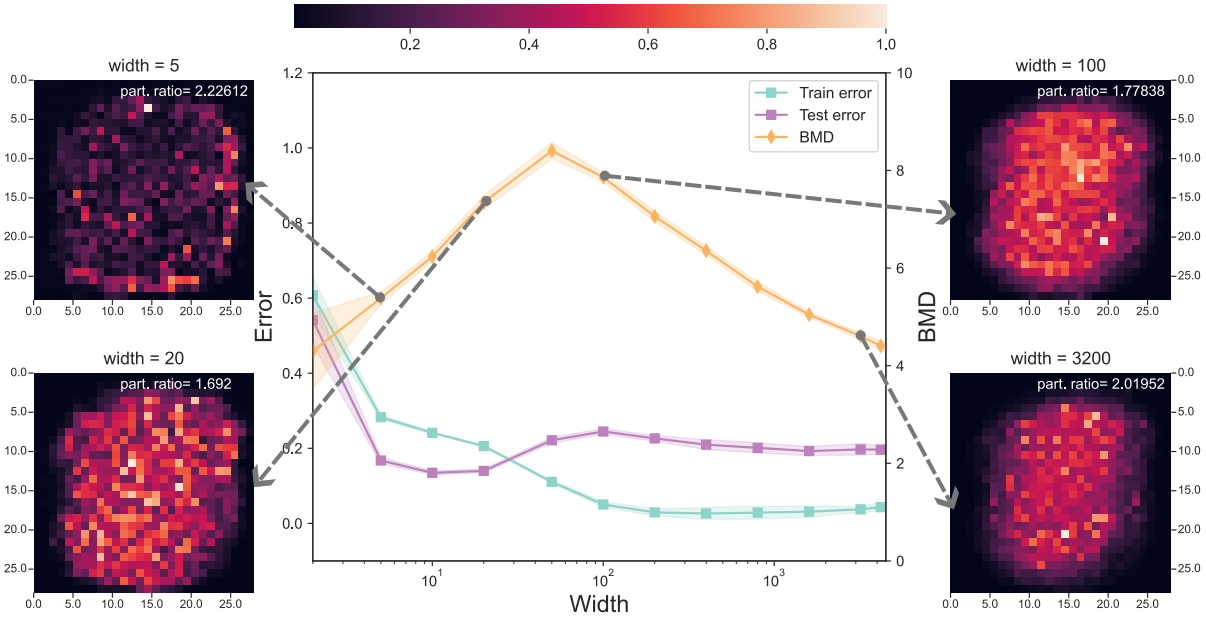

Figure 7: Heatmaps of the pixel contributions (τ_i^2 for $1 \leq i \leq 784$) estimated on the two-layer fully connected network trained on 20K samples of the MNIST dataset with 10 classes, 20% label noise and normalized to lie within $[0, 1]$ interval. The (rescaled) participation ratio is defined as $n \times \frac{\sum_{i=1}^{n} \tau_i^2}{(\sum_{i=1}^{n} \tau_i)^2}$. After the rescaling, a participation ratio of 1 indicates a uniform distribution of pixel contributions, while a value of n indicates a distribution concentrated over a single pixel. The heatmaps correspond to the contributions estimated with respect to label 0 for the models of different capacities (hidden layer dimensions) and represent only one seed, while the resulting curves on the plot represent an average over 20 different runs of the experiment.

5.6 Different Distributions for Estimating BMD

In BMD estimates for the previous experiments, Eq. (16), we focused on the case of i.i.d. binary input features. In the RFM, however, we have shown analytically that there exists a universality for the MD when one considers separable input distributions with the same first and second moments. In the numerical experiments, we have also shown evidence that the BMD peak can still provide insights into the behavior of the neural network function on the training and test data, which follow very different input statistics. To explore in detail the role of the input statistics, and of the presence of correlations in the input features, we measure the MD by resampling the inputs from different distributions: in Fig. 8 we plot the normalized MD curves for features sampled from:

- a uniform binary distribution (BMD).
- a standard normal (Gaussian) distribution $\mathcal{N}(0, 1)$.

- a uniform distribution in the range $[-1, 1]$.
- empirical distribution of the training data with random uniform resampling in the range $[-1, 1]$.

As one can see in Fig. 8, the MD curves estimated with binary and Gaussian i.i.d. inputs, with matching moments, are identical. With the uniform distribution, the second moment is $1/3$ and this results in a slightly rescaled MD curve. Introducing correlations in the inputs, in the MD estimated over the training data distribution, the curve still shows a similar behavior, and importantly the peak is found at the same value.

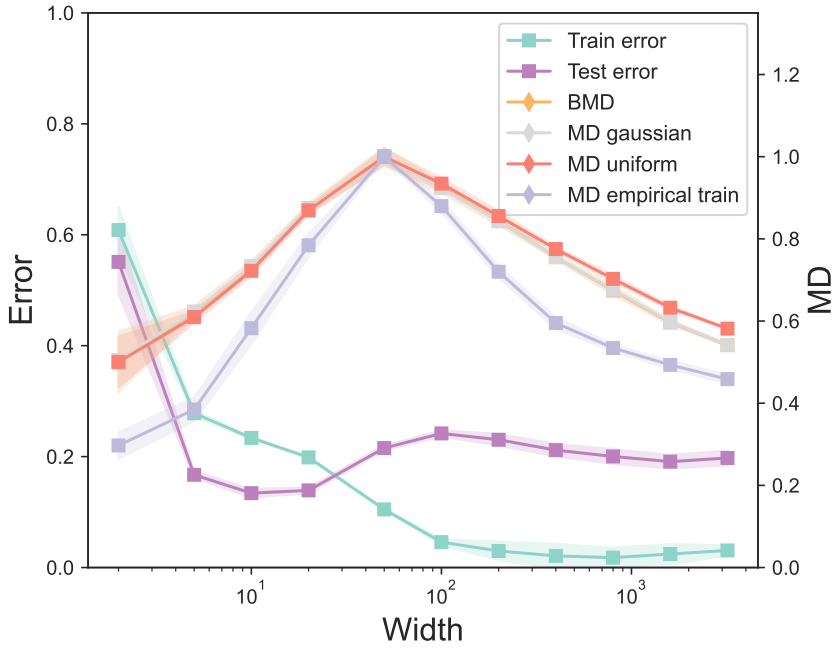

Figure 8: Mean dimensions estimated using Monte Carlo (eq. 16) w.r.t. different distributions of the two-layer fully connected network trained on the 20K of MNIST samples with 20% label noise and tested on 5K samples. The MD values are normalized to lie within [0,1] interval. The choice of the distribution does not affect the location of the peak. Moreover, distributions, which first two moments coincide (e.g. binary uniform Unif$\{-1, 1\}$ and Gaussian $\mathcal{N}(0, 1)$) yield the same MD pattern. The resulting plot represents an average over 20 different runs of the experiment.

6 Discussion

In this work, we analyzed the Boolean Mean Dimension as a tool for assessing the sensitivity of neural network functions. In the treatable setting of the Random Feature Model, we derived an exact characterization of the behavior of this metric as a function of the degree of overparameterization of the model. Notably, we found a strong correlation between the sharp increase of the BMD and the increase of the generalization error around the interpolation threshold. This finding indicates that as the neural network starts to overfit the noise in the data, the learned function becomes more sensitive to small perturbations of the input features. Importantly, while the double descent curve requires test data to be observed, the BMD can signal this type of failure mode by using information from the neural network alone. The same phenomenology appears in more realistic scenarios with different architectures and datasets, where factors influencing double descent, like regularization and label noise, are also found to

affect the BMD in similar fashion. Furthermore, we demonstrated that the BMD is informative about the vulnerability of trained models to adversarial attacks, despite assuming an input distribution that is very different from that of the training dataset.

Our study raises intriguing questions regarding the potential applications of BMD for regularization purposes. Another interesting future direction could be to investigate how comparing the BMDs achieved by a highly parametrized neural network trained on different datasets can help assess the effective dimensionality of the training data and the complexity of the discriminative tasks. Finally, it could be interesting to extend the study of the BMD in the RFM in the framework of a polynomial teacher model, recently analyzed in [Aguirre-López et al., 2024].

Acknowledgements

We thank Emanuele Borgonovo for the insightful discussions. Enrico Malatesta and Luca Saglietti aknowledge the MUR-Prin 2022 funding, Prot. 20229T9EAT and 2022E3WYTY, financed by the EU (Next Generation EU).

References

[Advani et al., 2020] Advani, M. S., Saxe, A. M., and Sompolinsky, H. (2020). High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446.

[Aguirre-López et al., 2024] Aguirre-López, F., Franz, S., and Pastore, M. (2024). Random features and polynomial rules. *arXiv preprint arXiv:2402.10164*.

[Baity-Jesi et al., 2018] Baity-Jesi, M., Sagun, L., Geiger, M., Spigler, S., Arous, G. B., Cammarota, C., LeCun, Y., Wyart, M., and Biroli, G. (2018). Comparing dynamics: Deep neural networks versus glassy systems. In *International Conference on Machine Learning*, pages 314–323. PMLR.

[Baldassi et al., 2022] Baldassi, C., Lauditi, C., Malatesta, E. M., Pacelli, R., Perugini, G., and Zecchina, R. (2022). Learning through atypical phase transitions in overparameterized neural networks. *Physical Review E*, 106(1):014116.

[Baldassi et al., 2020] Baldassi, C., Malatesta, E. M., Negri, M., and Zecchina, R. (2020). Wide flat minima and optimal generalization in classifying high-dimensional gaussian mixtures. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12):124012.

[Belkin et al., 2019] Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

[Brown et al., 2020] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

[d'Ascoli et al., 2020] d'Ascoli, S., Sagun, L., and Biroli, G. (2020). Triple descent and the two kinds of overfitting: Where & why do they appear? *Advances in Neural Information Processing Systems*, 33:3058–3069.

[d'Ascoli et al., 2020] d'Ascoli, S., Refinetti, M., Biroli, G., and Krzakala, F. (2020). Double trouble in double descent: Bias and variance (s) in the lazy regime. In *International Conference on Machine Learning*, pages 2280–2290. PMLR.

[Efron and Stein, 1981] Efron, B. and Stein, C. (1981). The jackknife estimate of variance. *The Annals of Statistics*, pages 586–596.

[Engel and Van den Broeck, 2001] Engel, A. and Van den Broeck, C. (2001). *Statistical mechanics of learning*. Cambridge University Press.

[Feinauer and Borgonovo, 2022] Feinauer, C. and Borgonovo, E. (2022). Mean dimension of generative models for protein sequences. *bioRxiv*, pages 2022–12.

[Geiger et al., 2019] Geiger, M., Spigler, S., d'Ascoli, S., Sagun, L., Baity-Jesi, M., Biroli, G., and Wyart, M. (2019). Jamming transition as a paradigm to understand the loss landscape of deep neural networks. *Physical Review E*, 100(1):012115.

[Gerace et al., 2020] Gerace, F., Loureiro, B., Krzakala, F., Mezard, M., and Zdeborova, L. (2020). Generalisation error in learning with random features and the hidden manifold model. In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3452–3462. PMLR.

[Gerace et al., 2022] Gerace, F., Saglietti, L., Mannelli, S. S., Saxe, A., and Zdeborová, L. (2022). Probing transfer learning with a model of synthetic correlated datasets. *Machine Learning: Science and Technology*, 3(1):015030.

[Glorot and Bengio, 2010] Glorot, X. and Bengio, Y. (2010). Xavier initialization. *J. Mach. Learn. Res.*

[Goldt et al., 2019] Goldt, S., Mézard, M., Krzakala, F., and Zdeborová, L. (2019). Modelling the influence of data structure on learning in neural networks: the hidden manifold model. *Physical Review X*, 10:041044.

[Guidotti et al., 2018] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42.

[Hahn et al., 2022] Hahn, R., Feinauer, C., and Borgonovo, E. (2022). The mean dimension of neural networks–what causes the interaction effects? *arXiv preprint arXiv:2207.04890*.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[Hoyt and Owen, 2021] Hoyt, C. and Owen, A. B. (2021). Efficient estimation of the anova mean dimension, with an application to neural net classification. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2):708–730.

[Jiang et al., 2019] Jiang, Y., Neyshabur, B., Mobahi, H., Krishnan, D., and Bengio, S. (2019). Fantastic generalization measures and where to find them. *arXiv preprint arXiv:1912.02178*.

[Jumper et al., 2021] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Liu and Owen, 2006] Liu, R. and Owen, A. B. (2006). Estimating mean dimensionality of analysis of variance decompositions. *Journal of the American Statistical Association*, 101(474):712–721.

[Liu et al., 2020] Liu, S., Papailiopoulos, D., and Achlioptas, D. (2020). Bad global minima exist and sgd can reach them. *Advances in Neural Information Processing Systems*, 33:8543–8552.

[Loureiro et al., 2021] Loureiro, B., Gerbelot, C., Cui, H., Goldt, S., Krzakala, F., Mézard, M., and Zdeborová, L. (2021). Capturing the learning curves of generic features maps for realistic data sets with a teacher-student model. *arXiv preprint arXiv:2102.08127*.

[Malatesta, 2023] Malatesta, E. M. (2023). High-dimensional manifold of solutions in neural networks: insights from statistical physics. *arXiv preprint arXiv:2309.09240*.

[Mei and Montanari, 2019] Mei, S. and Montanari, A. (2019). The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv preprint arXiv:1908.05355*.

[Mézard et al., 1987] Mézard, M., Parisi, G., and Virasoro, M. A. (1987). *Spin glass theory and beyond: An Introduction to the Replica Method and Its Applications*, volume 9. World Scientific Publishing Company.

[Mignacco et al., 2020] Mignacco, F., Krzakala, F., Lu, Y., Urbani, P., and Zdeborova, L. (2020). The role of regularization in classification of high-dimensional noisy gaussian mixture. In *International conference on machine learning*, pages 6874–6883. PMLR.

[Montavon et al., 2018] Montavon, G., Samek, W., and Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15.

[Nakkiran et al., 2021] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2021). Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003.

[Neyshabur et al., 2017] Neyshabur, B., Bhojanapalli, S., McAllester, D., and Srebro, N. (2017). Exploring generalization in deep learning. *Advances in neural information processing systems*, 30.

[Novak et al., 2018] Novak, R., Bahri, Y., Abolafia, D. A., Pennington, J., and Sohl-Dickstein, J. (2018). Sensitivity and generalization in neural networks: an empirical study. In *International Conference on Learning Representations*.

[O'Donnell, 2014] O'Donnell, R. (2014). *Analysis of boolean functions*. Cambridge University Press.

[OpenAI, 2023] OpenAI (2023). Gpt-4 technical report.

[Opper, 1995] Opper, M. (1995). Statistical mechanics of learning: Generalization. *The handbook of brain theory and neural networks*, pages 922–925.

[Owen, 2003] Owen, A. B. (2003). The dimension distribution and quadrature test functions. *Statistica Sinica*, pages 1–17.

[Pennington and Worah, 2017] Pennington, J. and Worah, P. (2017). Nonlinear random matrix theory for deep learning. *Advances in neural information processing systems*, 30.

[Rahimi et al., 2007] Rahimi, A., Recht, B., et al. (2007). Random features for large-scale kernel machines. In *NIPS*, volume 3, page 5. Citeseer.

[Ramesh et al., 2022] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical text-conditional image generation with clip latents, 2022. *URL https://arxiv.org/abs/2204.06125*, 7.

[Rombach et al., 2022] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. pages 10684–10695.

[Rudin, 2019] Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215.

[Sejnowski, 2020] Sejnowski, T. J. (2020). The unreasonable effectiveness of deep learning in artificial intelligence. *Proceedings of the National Academy of Sciences*, 117(48):30033–30038.

[Touvron et al., 2023] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models.

[Valle-Perez et al., 2018] Valle-Perez, G., Camargo, C. Q., and Louis, A. A. (2018). Deep learning generalizes because the parameter-function map is biased towards simple functions. *arXiv preprint arXiv:1805.08522*.

[Vapnik, 1999] Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Vilone and Longo, 2020] Vilone, G. and Longo, L. (2020). Explainable artificial intelligence: a systematic review. *arXiv preprint arXiv:2006.00093*.

Supplemental Material

A Proof of equation (15)

In order to prove the relation of the mean dimension of equation (15) in the main text, we anticipate two Lemmas that are useful for the proof.

Lemma A.1. *For any $u \subseteq [n]$, the ANOVA coefficients satisfy the following relation:*

$$f_u(x) = \sum_{v \subseteq u} (-1)^{|u|-|v|} \int f(\boldsymbol{x}) p(\boldsymbol{x}_{\setminus v} | \boldsymbol{x}_v) d\boldsymbol{x}_{\setminus v} = \sum_{v \subseteq u} (-1)^{|u|-|v|} \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_v] \tag{33}$$

Proof. This can be seen by induction. The base of induction for the empty set and a singleton $u = \{i\}, \forall i \in n$ are respectively

$$f_\emptyset = \mathbb{E}[f(\boldsymbol{x})] \tag{34a}$$
$$f_{\{i\}} = \mathbb{E}[f(\boldsymbol{x})|x_i] - \mathbb{E}[f(\boldsymbol{x})] = \mathbb{E}[f(\boldsymbol{x})|x_i] - f_\emptyset \tag{34b}$$

having used the ANOVA recursion in equation (2). Assuming equation (33) holds for the sets up to the size k we can show the induction step up to the set size $k+1$. Given a set $S \subseteq [n]$, $i \notin S$ and $|S| = k$ denoting $u = S \cup \{i\}$ we have:

$$\begin{aligned} f_u(\boldsymbol{x}_u) &= \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_u] - \sum_{v \subset u} f_v(\boldsymbol{x}_v) = \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_u] - \sum_{v \subset u} \sum_{t \subseteq v} (-1)^{|v|-|t|} \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_t] \\ &= \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_u] - \sum_{t \subseteq S} (-1)^{-|t|} \sum_{t \subseteq v \subset u} (-1)^{|v|} \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_t] \,. \end{aligned} \tag{35}$$

The summation over sets $t \subseteq v \subset u$ can be performed

$$\begin{aligned} \sum_{t \subseteq v \subset u} (-1)^{|v|} &= \sum_{k=|t|}^{|u|-1} (-1)^k \binom{|u|-|t|}{k-|t|} \\ &= (-1)^{-|t|} \sum_{k=0}^{|u|-|t|-1} (-1)^k \binom{|u|-|t|}{k} = (-1)^{|u|+1} \,. \end{aligned} \tag{36}$$

Inserting this back into (35) we have finally

$$f_u(\boldsymbol{x}_u) = \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_u] + \sum_{t \subseteq S} (-1)^{|u|-|t|} \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_t] = \sum_{v \subseteq u} (-1)^{|u|-|v|} \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{x}_v] \tag{37}$$

□

Now given the previous Lemma A.1 we can show:

Lemma A.2. *We can write*

$$f(\boldsymbol{x}) = \sum_{u \ni i} f_u(\boldsymbol{x}_u) + \mathbb{E}\left[f(\boldsymbol{x})|\boldsymbol{x}_{\setminus i}\right] \tag{38}$$

or equivalently

$$\sum_{u \not\ni i} f_u(\boldsymbol{x}_u) = \mathbb{E}\left[f(\boldsymbol{x})|\boldsymbol{x}_{\setminus i}\right] \tag{39}$$

Proof. To show that we consider:

$$\sum_{u\ni i} f_u(\boldsymbol{x}_u) = \sum_{u\ni i}\sum_{v\subseteq u}(-1)^{|u|-|v|}\mathbb{E}\left[f(\boldsymbol{x})|\boldsymbol{x}_v\right] = \sum_v \left[\sum_{u\supseteq v, u\ni i}(-1)^{|u|}\right](-1)^{-|v|}\mathbb{E}\left[f(\boldsymbol{x})|\boldsymbol{x}_v\right] \tag{40}$$

The term in the squared brackets can be computed. We need to distinguish two cases, i.e. $i \in v$, and $i \notin v$. In the first case we get

$$\sum_{u\supseteq v, u\ni i}(-1)^{|u|} = \sum_{k=|v|}^{n}(-1)^k\binom{n-|v|}{k-|v|} = \sum_{k=0}^{n-|v|}(-1)^{k+|v|}\binom{n-|v|}{k} = (-1)^{|v|}\delta_{n,|v|} \tag{41}$$

where $\delta_{i,j}$ is the Kronecker delta function. If $i \notin v$, instead

$$\sum_{u\supseteq v, u\ni i}(-1)^{|u|} = \sum_{k=|v|+1}^{n}(-1)^k\binom{n-|v|-1}{k-|v|-1} = (-1)^{|v|+1}\delta_{n,|v|+1} \tag{42}$$

i.e. we get a non-zero result only if $v = [n]$ or $v = [n]\backslash\{i\}$. Therefore we get

$$\sum_{u\ni i} f_u(\boldsymbol{x}_u) = f(\boldsymbol{x}) - \mathbb{E}\left[f(\boldsymbol{x})|\boldsymbol{x}_{\backslash i}\right]$$

□

In the following we will denote by $\boldsymbol{x}^{\oplus i}$ a vector $\boldsymbol{x}$ with a resampled i_{th} coordinate. We are now ready to prove the following theorem:

Theorem A.3. *The mean dimension can be written as*

$$M_f = \sum_{u\subseteq[n]}|u|\frac{\sigma_u^2}{\sigma^2} = \frac{\sum_{i=1}^n \tau_i^2}{\sigma^2} \tag{43}$$

where

$$\tau_i^2 = \frac{1}{2}\int d\boldsymbol{x}\, dx_i'\, p(\boldsymbol{x})\, p(x_i'|\boldsymbol{x}_{\backslash i})(f(\boldsymbol{x}) - f(\boldsymbol{x}^{\oplus i}))^2 \equiv \frac{1}{2}\mathbb{E}\left(f(\boldsymbol{x}) - f(\boldsymbol{x}^{\oplus i})\right)^2 \tag{44}$$

Proof. We can write the numerator of the mean dimension as

$$\sum_{u\subseteq[n]}|u|\sigma_u^2 = \sum_{k=1}^{n} k \sum_{\substack{u\subseteq[n]\\|u|=k}}\sigma_u^2 = \sum_{i=1}^{n}\sum_{u\subseteq[n]:i\in u}\sigma_u^2\,. \tag{45}$$

In the first equality we divided the summation over the sets into a double summation over the size k of the set and a summation over the sets of fixed size k. In the second equality we have used the fact that the summation over k can be interpreted as a summation over the indices i of the variable $\boldsymbol{x}$; the inner sum can be therefore written as a summation over the sets (of any possible size) that contain the variable i itself. Reminding that

$$\sigma_u^2 \equiv \int f_u^2(\boldsymbol{x}_u)\, p_u(\boldsymbol{x}_u)\, d\boldsymbol{x}_u = \mathbb{E}[f_u^2(\boldsymbol{x}_u)]\,, \tag{46}$$

and using Lemma A.2 and orthogonality of the coefficients of the ANOVA expansion, we have

$$\begin{aligned}
\tau_i^2 \equiv \frac{1}{2}\mathbb{E}\left(f(\boldsymbol{x}) - f(\boldsymbol{x}^{\oplus i})\right)^2 &= \frac{1}{2}\mathbb{E}\left(f(\boldsymbol{x}) - f(\boldsymbol{x}^{\oplus i})\right)\left(\sum_{u\ni i} f_u(\boldsymbol{x}_u) - \sum_{u\ni i} f_u(\boldsymbol{x}_u^{\oplus i})\right)\\
&= \mathbb{E}\left[f(\boldsymbol{x})\sum_{u\ni i} f_u(\boldsymbol{x}_u)\right] - \mathbb{E}\left[f(\boldsymbol{x})\sum_{i\ni i} f_u(\boldsymbol{x}_u^{\oplus i})\right]\\
&= \mathbb{E}\left[\sum_{u\ni i} f_u^2(\boldsymbol{x}_u)\right] - \mathbb{E}\left[f(\boldsymbol{x})\left(f(\boldsymbol{x}^{\oplus i}) - \mathbb{E}\left[f(\boldsymbol{x}^{\oplus i}|\boldsymbol{x}_{\backslash i})\right]\right)\right]\\
&= \mathbb{E}\sum_{u\ni i} f_u^2(\boldsymbol{x}_u) - 0 = \sum_{u\ni i}\sigma_u^2\,.
\end{aligned}$$

□

B Replica computation of the Boolean Mean Dimension in the random feature model

B.1 Free entropy

We review here the replica calculations of the free entropy of the model, defined as

$$\phi_\beta = \lim_{N\to\infty} \frac{1}{N} \mathbb{E}_{\mathcal{D},F} \ln Z_\beta \,. \tag{47}$$

This in turn will give the information necessary to compute the BMD. The average over the dataset in (47) can be performed using the replica trick

$$\mathbb{E}_{\mathcal{D},F} \ln Z_\beta = \lim_{n\to 0} \frac{1}{n} \ln \left(\mathbb{E}_{\mathcal{D},F} Z_\beta^n \right) \,. \tag{48}$$

In the following we will consider n as an integer and we will denote by a, b as replica indices running from 1 to n.

Gaussian equivalence theorem In order to compute the average over the input patterns of Z_β^n, we will apply a central limit theorem [Pennington and Worah, 2017], valid in the thermodynamic limit where N, D, P go to infinity with fixed $\alpha \equiv \frac{P}{N}$ and $\alpha_D \equiv \frac{D}{N}$. In the statistical physics literature this central limit is often called Gaussian equivalence theorem (GET) [Goldt et al., 2019, Loureiro et al., 2021]. It can indeed be shown that the model is *equivalent* to a Gaussian covariate model [Mei and Montanari, 2019] and the following identification can be made

$$\sigma\left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki}\, x_k \right) = \kappa_0 + \frac{\kappa_1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki}\, x_k + \kappa_\star \eta_i \tag{49}$$

where η_i is Gaussian noise with zero mean and unit variance and we have defined the following coefficients

$$\kappa_0 = \int Dz\, \sigma(z) \tag{50a}$$

$$\kappa_1 = \int Dz\, z\, \sigma(z) = \int Dz\, \sigma'(z) \tag{50b}$$

$$\kappa_2 = \int Dz\, \sigma^2(z) \tag{50c}$$

$$\kappa_\star^2 = \kappa_2 - \kappa_1^2 - \kappa_0^2 \tag{50d}$$

with $Dz \equiv \frac{e^{-z^2/2}}{\sqrt{2\pi}} dz$. In the following we will considered for simplicity $\sigma(\cdot)$ to be an odd activation function, so that in this case $\kappa_0 = 0$.

Average over the dataset Using GET, we therefore get

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}} Z_\beta^n &= \mathbb{E}_{\boldsymbol{w}^T} \int \prod_a d\boldsymbol{w}^a \int \prod_\mu \frac{du^\mu d\hat{u}^\mu}{2\pi} \prod_{\mu a} \frac{d\lambda_a^\mu d\hat{\lambda}_a^\mu}{2\pi}\, e^{-\beta \sum_{\mu a} \ell\left(\mathrm{sign}(u^\mu)\lambda_a^\mu\right) - \frac{\beta\lambda}{2} \sum_{ia} (w_i^a)^2} \\
&\quad \times \prod_\mu e^{i u^\mu \hat{u}^\mu + i \sum_a \lambda_a^\mu \hat{\lambda}_a^\mu - \frac{(\hat{u}^\mu)^2}{2} - \frac{1}{2} \sum_{ab} Q_{ab} \hat{\lambda}_a^\mu \hat{\lambda}_b^\mu - \sum_a M_a \hat{u}^\mu \hat{\lambda}_a^\mu} \,. \\
&= \int \prod_a d\boldsymbol{w}^a \int \prod_\mu du^\mu \prod_{\mu a} d\lambda_a^\mu\, e^{-\beta \sum_{\mu a} \ell\left(\mathrm{sign}(u^\mu)\lambda_a^\mu\right) - \frac{\beta\lambda}{2} \sum_{ia} (w_i^a)^2} \prod_\mu \mathcal{N}(u^\mu, \lambda_a^\mu; \mathbf{0}, \Sigma_{ab})
\end{aligned} \tag{51}$$

where the correlation matrix of the $n+1$-dimensional multivariate Gaussian $\mathcal{N}$ is

$$\Sigma_{ab} \equiv \begin{pmatrix} \rho & M_a \\ M_a & Q_{ab}\,. \end{pmatrix} \tag{52}$$

Here $\rho = \frac{1}{D}\sum_{k=1}^{D}(w_k^T)^2 = 1$, since the teacher has fixed norm. In the previous equation we have also denoted by Q_{ab} and M_a the following quantities

$$M_a = \kappa_1 \frac{1}{D}\sum_{k=1}^{D} s_k^a w_k^T \equiv \kappa_1 r_a \tag{53a}$$

$$Q_{ab} = \kappa_\star^2 \frac{1}{N}\sum_{i=1}^{N} w_i^a w_i^b + \kappa_1^2 \frac{1}{D}\sum_{k=1}^{D} s_k^a s_k^b \equiv \kappa_\star^2 q_{ab} + \kappa_1^2 p_{ab} \tag{53b}$$

where we have defined the projected student weights in the space of the teacher s_k^a as

$$s_k^a \equiv \frac{1}{\sqrt{N}}\sum_{i=1}^{N} F_{ki} w_i^a\,. \tag{54}$$

Average over Gaussian Features We can enforce the definition of the projected weights in (54) using delta functions and their integral representation. It then becomes easy to perform the average over random Gaussian features. We get a terms of the following form

$$\int \prod_{ka} \frac{ds_k^a d\hat{s}_k^a}{2\pi}\, e^{i\sum_{ka} s_k^a \hat{s}_k^a} \prod_{ki} \mathbb{E}_{F_{ki}} \left[e^{-i\frac{F_{ki}}{\sqrt{N}}\sum_a \hat{s}_k^a w_i^a} \right] = \int \prod_{ka} \frac{ds_k^a d\hat{s}_k^a}{2\pi}\, e^{i\sum_{ka} s_k^a \hat{s}_k^a - \frac{1}{2}\sum_{ab,k} \hat{s}_k^a \hat{s}_k^b \left(\frac{1}{N}\sum_i w_i^a w_i^b\right)}\,, \tag{55}$$

which only depends on the q_{ab} defined in (53b).

Saddle point method We can now impose the definitions of the order parameters

$$q_{ab} \equiv \frac{1}{N}\sum_i w_i^a w_i^b\,, \qquad p_{ab} \equiv \frac{1}{D}\sum_k s_k^a s_k^b\,, \qquad r_a \equiv \frac{1}{D}\sum_k s_k^a w_k^T\,. \tag{56}$$

Denoting by $\langle\cdot\rangle_{\mathcal{D},F}$ the average over both patterns and random features, the final result reads

$$\mathbb{E}_{\mathcal{D},F} Z_\beta^n = \int \prod_{a\le b} \frac{dq_{ab} d\hat{q}_{ab}}{2\pi} \prod_{a\le b} \frac{dp_{ab} d\hat{p}_{ab}}{2\pi} \prod_a \frac{dr_a d\hat{r}_a}{2\pi}\, e^{N\phi_\beta^{(n)}} \tag{57}$$

where

$$\phi_\beta^{(n)} = -\frac{1}{2}\sum_{ab} q_{ab}\hat{q}_{ab} - \frac{\alpha_D}{2}\sum_{ab} p_{ab}\hat{p}_{ab} - \alpha_D \sum_a r_a \hat{r}_a + G_{SS} + \alpha_D G_{SE} + \alpha G_E \tag{58a}$$

$$G_{SS} = \ln \int \prod_a dw_a\, e^{\frac{1}{2}\sum_{ab}\hat{q}_{ab} w_a w_b - \frac{\beta\lambda}{2}\sum_a w_a^2} \tag{58b}$$

$$G_{SE} = \ln \int \prod_a \frac{ds_a d\hat{s}_a}{2\pi}\, e^{i\sum_a s_a \hat{s}_a + \sum_a \hat{r}_a s_a + \frac{1}{2}\sum_{ab}\hat{p}_{ab} s_a s_b - \frac{1}{2}\sum_{ab} q_{ab}\hat{s}_a\hat{s}_b} \tag{58c}$$

$$G_E = \ln \int \prod_a \frac{d\lambda_a d\hat{\lambda}_a}{2\pi} \frac{du d\hat{u}}{2\pi}\, e^{iu\hat{u} + i\sum_a \lambda_a\hat{\lambda}_a - \beta\sum_a \ell(\mathrm{sign}(u)\lambda_a) - \frac{\hat{u}^2}{2} - \frac{1}{2}\sum_{ab} Q_{ab}\hat{\lambda}_a\hat{\lambda}_b - \hat{u}\sum_a M_a\hat{\lambda}_a} \tag{58d}$$

and M_a, Q_{ab} are defined in terms of q_{ab}, p_{ab}, r_a in (53). Notice that the integrals inside the "entropic" G_S and the G_{SE} tersm can be solved analytically by using multivariate Gaussian integrals identities.

Replica Symmetric Ansatz We focus on the saddle points that preserve the symmetry between the exchange of replicas (RS)

$$r_a = r\,, \qquad \hat{r}_a = \hat{r}\,, \qquad \forall a \in [n] \tag{59a}$$

$$q_{aa} = q_d\,, \qquad \hat{q}_{aa} = -\hat{q}_d\,, \qquad p_{aa} = p_d\,, \qquad \hat{p}_{aa} = -\hat{p}_d\,, \qquad \forall a \in [n] \tag{59b}$$

$$q_{ab} = q\,, \qquad \hat{q}_{ab} = \hat{q}\,, \qquad p_{ab} = p\,, \qquad \hat{p}_{ab} = \hat{p}\,, \qquad 1 \le a \le b \le n \tag{59c}$$

Denoting with

$$M \equiv \kappa_1 r\,, \tag{60a}$$

$$Q \equiv \kappa_\star^2 q + \kappa_1^2 p\,, \tag{60b}$$

$$Q_d \equiv \kappa_\star^2 q_d + \kappa_1^2 p_d \tag{60c}$$

we finally get in the small n limit the free entropy reads

$$\phi_\beta \equiv \lim_{n\to 0} \frac{\phi_\beta^{(n)}}{n} = \frac{1}{2}(q_d\hat{q}_d + q\hat{q}) + \frac{\alpha_D}{2}\,(p_d\hat{p}_d + p\hat{p}) - \alpha_D r\hat{r} + \mathcal{G}_{SS} + \alpha_D \mathcal{G}_{SE} + \alpha \mathcal{G}_E\,, \tag{61}$$

where

$$\mathcal{G}_{SS} \equiv \lim_{n\to 0} \frac{G_{SS}}{n} = \frac{1}{2}\ln\left(\frac{2\pi}{\hat{q}_d + \hat{q} + \beta\lambda}\right) + \frac{\hat{q}}{2(\hat{q}_d + \hat{q} + \beta\lambda)} \tag{62a}$$

$$\mathcal{G}_{SE} \equiv \lim_{n\to 0} \frac{G_{SE}}{n} = -\frac{q}{2(q_d - q)} - \frac{1}{2}\ln\left[1 + (\hat{p} + \hat{p}_d)(q_d - q)\right] + \frac{1}{2}\frac{(\hat{p} + \hat{r}^2)(q_d - q) + \frac{q}{q_d - q}}{1 + (\hat{p} + \hat{p}_d)(q_d - q)} \tag{62b}$$

$$\mathcal{G}_E \equiv \lim_{n\to 0} \frac{G_E}{n} = 2\int Dz_0\, H\left(-\frac{M z_0}{\sqrt{Q - M^2}}\right) \ln \int Dz_1\, e^{-\beta\ell\left(\sqrt{Q} z_0 + \sqrt{Q_d - Q} z_1\right)} \tag{62c}$$

and $H(x) = \int_x^\infty Dz = \frac{1}{2}\mathrm{Erfc}\left(\frac{x}{\sqrt{2}}\right)$. The values of the 10 order parameters, namely q, $\hat{q}$, q_d, $\hat{q}_d$, p, $\hat{p}$, p_d, $\hat{p}_d$, r, $\hat{r}$ must be found self-consistently by solving the saddle point equations obtained by differentiating ϕ.

Large β limit We now restrict the discussion to the case of the convex loss functions as the MSE and the CE loss cited in the main text (23). In this case, we have to impose the following scalings

$$\hat{r} \to \beta\hat{r}\,, \tag{63a}$$

$$q = q_d - \frac{\delta q}{\beta}\,, \qquad p = p_d - \frac{\delta p}{\beta}\,, \tag{63b}$$

$$\hat{q} = \beta^2 \delta\hat{q} + \frac{\beta}{2}\delta\hat{Q}\,, \qquad \hat{q}_d = -\beta^2\delta\hat{q} + \frac{\beta}{2}\delta\hat{Q}\,, \tag{63c}$$

$$\hat{p} = \beta^2 \delta\hat{p} + \frac{\beta}{2}\delta\hat{P}\,, \qquad \hat{p}_d = -\beta^2\delta\hat{p} + \frac{\beta}{2}\delta\hat{P}\,. \tag{63d}$$

We have that the free energy which was defined in the main text in equation (26) is

$$-f = \lim_{\beta\to\infty} \frac{\phi_\beta}{\beta} = \frac{1}{2}\left(q_d\delta\hat{Q} - \delta q\delta\hat{q}\right) + \frac{\alpha_D}{2}\left(p_d\delta\hat{P} - \delta p\delta\hat{p}\right) - \alpha_D r\hat{r} + \mathcal{G}_{SS} + \alpha_D\mathcal{G}_{SE} + \alpha\mathcal{G}_E\,, \tag{64}$$

where

$$\mathcal{G}_{SS} = \frac{\delta\hat{q}}{2(\delta\hat{Q} + \lambda)} \tag{65a}$$

$$\mathcal{G}_{SE} = -\frac{q_d}{2\delta q} + \frac{1}{2}\frac{(\delta\hat{p} + \hat{r}^2)\,\delta q + \frac{q_d}{\delta q}}{1 + \delta\hat{P}\,\delta q} \tag{65b}$$

$$\mathcal{G}_E = 2\int Dz_0\, H\left(-\frac{M z_0}{\sqrt{Q_d - M^2}}\right) \max_{z_1}\left[-\frac{z_1^2}{2} - \ell(\sqrt{Q_d} z_0 + \sqrt{\delta Q} z_1)\right]\,. \tag{65c}$$

We have also denoted by $\delta Q = \kappa_\star^2 \delta q + \kappa_1^2 \delta p$. Again the order parameters δq, $\delta \hat{q}$, q_d, $\delta \hat{Q}$, δp, $\delta \hat{p}$, p_d, $\delta \hat{P}$, r, $\hat{r}$ must be found self-consistently by solving the saddle point equations obtained by differentiating f.

Physical observables of interest As shown in multiple papers, see e.g. [Gerace et al., 2020, Baldassi et al., 2022] the generalization error can be obtained by computing

$$\epsilon_g = \frac{1}{\pi} \arccos\left(\frac{M}{\sqrt{Q_d}}\right) . \tag{66}$$

The training loss can be computed by a derivative with respect to β

$$\ell_t = \lim_{\beta\to\infty} \frac{\partial(\beta f)}{\partial \beta} = 2 \int Dz_0 \, H\left(-\frac{M z_0}{\sqrt{Q_d - M^2}}\right) \ell\left(\sqrt{Q_d} z_0 + \sqrt{\delta Q} z_1^\star\right) \tag{67}$$

with

$$z_1^\star = \arg\max_{z_1} \left[-\frac{z_1^2}{2} - \ell(\sqrt{Q_d} z_0 + \sqrt{\delta Q} z_1)\right] \tag{68}$$

The test loss can be computed as follows [Baldassi et al., 2022]

$$\begin{aligned} \ell_g &\equiv \langle \mathbb{E}_{\boldsymbol{\xi}^\star} \ell\left(y^\star \hat{y}^\star\right)\rangle \\ &= \int \frac{du d\hat{u}}{2\pi} \frac{d\lambda d\hat{\lambda}}{2\pi} \, \ell\left(\mathrm{sign}(u)\lambda\right) e^{iu\hat{u} + i\lambda\hat{\lambda} - \frac{\hat{u}^2}{2} - \frac{1}{2} Q_d \hat{\lambda}^2 - M\hat{u}\hat{\lambda}} \\ &= 2 \int Dv \, \ell\left(-\sqrt{Q_d} v\right) H\left(-\frac{Mv}{\sqrt{Q_d - M^2}}\right) . \end{aligned} \tag{69}$$

where $\boldsymbol{\xi}^\star$ represents a new extracted pattern, $y^\star$ its corresponding label and $\hat{y}^\star$ the prediction of the model.

B.2 Analytical determination of the BMD

In the following subsections we will show the derivation of the annealed averages on the inputs of the numerator and denominator of the BMD.

B.2.1 Annealed average of the denominator of the BMD

The denominator in the definition of the BMD is easy to analyze. Applying GET, we obtain

$$\begin{aligned} \left\langle \hat{y}^2(\boldsymbol{w}; \boldsymbol{x}) \right\rangle &= \left\langle \frac{1}{N} \sum_{i,j} w_i w_j \, \sigma\left(\frac{1}{\sqrt{D}} \sum_{k=1}^D F_{ki} \, x_k\right) \sigma\left(\frac{1}{\sqrt{D}} \sum_{k=1}^D F_{kj} \, x_k\right) \right\rangle \\ &= \left\langle \frac{1}{N} \sum_{i,j} w_i w_j \left(\kappa_0 + \frac{\kappa_1}{\sqrt{D}} \sum_{k=1}^D F_{ki} \, x_k + \kappa_\star \eta_i\right) \left(\kappa_0 + \frac{\kappa_1}{\sqrt{D}} \sum_{k=1}^D F_{kj} \, x_k + \kappa_\star \eta_j\right) \right\rangle \\ &= \left(\frac{\kappa_0}{\sqrt{N}} \sum_i w_i\right)^2 + \frac{\kappa_1^2}{D} \sum_k \left(\frac{1}{\sqrt{N}} \sum_i F_{ki} w_i\right) \left(\frac{1}{\sqrt{N}} \sum_j F_{kj} w_j\right) + \frac{\kappa_\star^2}{N} \sum_i w_i^2 = Q_d \end{aligned} \tag{70}$$

The average squared is instead simply given by

$$\langle \hat{y}(\boldsymbol{w}; \boldsymbol{x}) \rangle^2 = \left(\frac{\kappa_0}{\sqrt{N}} \sum_i w_i\right)^2 \tag{71}$$

Therefore the denominator in the BMD reads

$$\left\langle \hat{y}^2(\boldsymbol{w};\boldsymbol{x}) \right\rangle - \langle \hat{y}(\boldsymbol{w};\boldsymbol{x}) \rangle^2 = \frac{\kappa_1^2}{D} \sum_k \left(\frac{1}{\sqrt{N}} \sum_i F_{ki} w_i \right) \left(\frac{1}{\sqrt{N}} \sum_j F_{kj} w_j \right) + \frac{\kappa_\star^2}{N} \sum_i w_i^2 = Q_d \quad (72)$$

where Q_d is the overlap obtained by the RS saddle point equations sketched in the previous section.

B.2.2 Annealed average of the numerator of the BMD

More work has to be done to compute the numerator of the BMD. We can write it as

$$\begin{aligned}
&\sum_{k'=1}^{D} \left\langle (\hat{y}(\boldsymbol{w};\boldsymbol{x}) - \hat{y}(\boldsymbol{w};\boldsymbol{x}^{\oplus k'}))^2 \right\rangle = \\
&= \sum_{k'=1}^{D} \left\langle \left(\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \, \sigma \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k \right) - \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \, \sigma \left(\frac{1}{\sqrt{D}} \sum_{k \neq k'} F_{ki} \, x_k + \frac{F_{k'i} \tilde{x}_{k'}}{\sqrt{D}} \right) \right)^2 \right\rangle \\
&= \sum_{k'=1}^{D} \left\langle \left(\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \, \sigma \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k \right) - \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \, \sigma \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k - \frac{F_{k'i}(x_{k'} - \tilde{x}_{k'})}{\sqrt{D}} \right) \right)^2 \right\rangle
\end{aligned} \quad (73)$$

Taylor expanding the second term we have

$$\begin{aligned}
&\frac{1}{2} \sum_{k'=1}^{D} \left\langle (\hat{y}(\boldsymbol{w};\boldsymbol{x}) - \hat{y}(\boldsymbol{w};\boldsymbol{x}^{\oplus k'}))^2 \right\rangle \\
&= \frac{1}{2} \sum_{k'=1}^{D} \left\langle \left(\frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i \, \sigma' \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k \right) \frac{F_{k'i}(x_{k'} - \tilde{x}_{k'})}{\sqrt{D}} \right)^2 \right\rangle \\
&= \frac{1}{2} \sum_{k'=1}^{D} \left\langle \frac{1}{N} \sum_{i,j} w_i w_j \, \sigma' \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k \right) \sigma' \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{kj} \, x_k \right) \frac{F_{k'i} F_{k'j} (x_{k'} - \tilde{x}_{k'})^2}{D} \right\rangle .
\end{aligned} \quad (74)$$

We then apply the GET and we take the average over the inputs getting

$$\begin{aligned}
&\frac{1}{2} \left\langle \sigma' \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k \right) \sigma' \left(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} F_{kj} \, x_k \right) (x_{k'} - \tilde{x}_{k'})^2 \right\rangle \\
&= \frac{1}{2} \left\langle \left[\bar{\kappa}_0 + \frac{\bar{\kappa}_1}{\sqrt{D}} \sum_{k=1}^{D} F_{ki} \, x_k + \bar{\kappa}_\star \eta_i \right] \left[\bar{\kappa}_0 + \frac{\bar{\kappa}_1}{\sqrt{D}} \sum_{k=1}^{D} F_{kj} \, x_k + \bar{\kappa}_\star \eta_j \right] (x_{k'} - \tilde{x}_{k'})^2 \right\rangle \\
&= \bar{\kappa}_0^2 + \bar{\kappa}_\star^2 \delta_{ij} + \frac{\bar{\kappa}_1^2}{2D} \sum_{k=1}^{D} \sum_{k''=1}^{D} F_{ki} \, F_{k''j} \left\langle x_k x_{k''} (x_{k'} - \tilde{x}_{k'})^2 \right\rangle \\
&= \bar{\kappa}_0^2 + \bar{\kappa}_\star^2 \delta_{ij} + \frac{\bar{\kappa}_1^2}{D} \left(2 F_{k'i} \, F_{k'j} + \sum_k F_{ki} \, F_{kj} \right) = \bar{\kappa}_0^2 + \bar{\kappa}_\star^2 \delta_{ij} + \frac{\bar{\kappa}_1^2}{D} \sum_k F_{ki} \, F_{kj} \, .
\end{aligned} \quad (75)$$

where we have denoted for simplicity by $\bar{\kappa}_0$, $\bar{\kappa}_1$, $\bar{\kappa}_2$, $\bar{\kappa}_\star$ the coefficients in equation (50) for σ'. Notice that in all the steps that we have done to arrive performing the average over the inputs we have not used anywhere the binary nature of the inputs. It is therefore easy to see that we would have obtained the same result if we had chosen a probability distribution on the inputs with the same first two moments (e.g., a standard normal distribution). Inserting this back into

the previous equation, we get

$$\begin{aligned}
&\frac{1}{2}\sum_{k'=1}^{D}\left\langle(\hat{y}(\boldsymbol{w};\boldsymbol{x})-\hat{y}(\boldsymbol{w};\boldsymbol{x}^{\oplus k'}))^2\right\rangle \\
&=\sum_{k'=1}^{D}\frac{1}{N}\sum_{ij}w_iw_j\left[\bar{\kappa}_0^2+\bar{\kappa}_\star^2\delta_{ij}+\frac{\bar{\kappa}_1^2}{D}\sum_k F_{ki}\,F_{kj}\right]\frac{F_{k'i}F_{k'j}}{D} \\
&=\frac{\bar{\kappa}_0^2}{N}\sum_{ij}\Omega_{ij}w_iw_j+\frac{\bar{\kappa}_\star^2}{N}\sum_i\Omega_{ii}w_i^2+\frac{\bar{\kappa}_1^2}{N}\sum_{ij}\Omega_{ij}^2w_iw_j=\frac{1}{N}\sum_{ij}\bar{\Psi}_{ij}w_iw_j
\end{aligned} \tag{76}$$

where we have introduced

$$\bar{\Psi}_{ij}\equiv\bar{\kappa}_\star^2\,\Omega_{ii}\delta_{ij}+\bar{\kappa}_0^2\,\Omega_{ij}+\bar{\kappa}_1^2\,\Omega_{ij}^2\,, \tag{77a}$$

$$\Omega_{ij}\equiv\frac{1}{D}\sum_{k=1}^{D}F_{ki}F_{kj}\,. \tag{77b}$$

Therefore the mean dimension for a generic non-linearity σ is

$$M_f(\boldsymbol{w})=\frac{\frac{1}{N}\sum_{ij}\bar{\Psi}_{ij}w_iw_j}{Q_d} \tag{78}$$

B.2.3 BMD for an odd non-linearity σ

If we assume the activation function to be odd we have $\bar{\kappa}_1=0$, and $\bar{\kappa}_0=\kappa_1$. Therefore we can write the BMD in terms of the order parameters only

$$M_f\equiv\langle M_f(\boldsymbol{w})\rangle_{\boldsymbol{w}}=\frac{Q_d+(\bar{\kappa}_\star^2-\kappa_\star^2)\,q_d}{Q_d} \tag{79}$$

where

$$\bar{\kappa}_\star^2-\kappa_\star^2=\int Dz\,(\sigma'(z))^2-\int Dz\,\sigma^2(z)=\bar{\kappa}_2-\kappa_2 \tag{80}$$

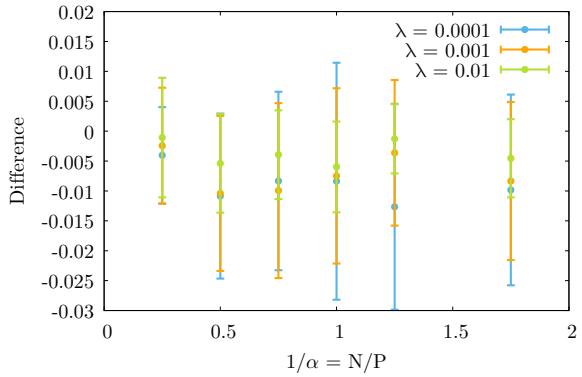

Figure 9: Difference between the BMD estimated using Monte Carlo method and using equation (79).

Notice that equation (79) can be used as an alternative (very efficient) way of computing the BMD, without using Monte Carlo method. We show in Fig. 9 how the difference between the BMD estimated using Monte Carlo and the one using equation (79) showing good agreement.

In the limit of $1/\alpha \to 0$, at fixed $\alpha_T = P/D$ (i.e. $N \to 0$), the solution to the saddle point equations show that $p_d \to q_d$; in this limit the BMD goes to

$$M_f = 1 + \frac{\bar{\kappa}_\star^2 - \kappa_\star^2}{\kappa_\star^2 + \kappa_1^2} = \frac{\bar{\kappa}_2}{\kappa_2} \,. \tag{81}$$

For example for $\sigma(x) = \tanh(x)$ activation we get $M_f = 1.1778$ as it is displayed in Fig. 1.

C Double peak behavior of the BMD

As can be seen in Fig. 11, if α_T is sufficiently large the BMD can display a peak in addition to the one located at the interpolation threshold ($N = P$ for the MSE loss). This secondary peak is located at $\alpha = \alpha_T$, i.e. when the number of parameters is equal to the input dimension $N = D$. This peak is not present in the generalization.

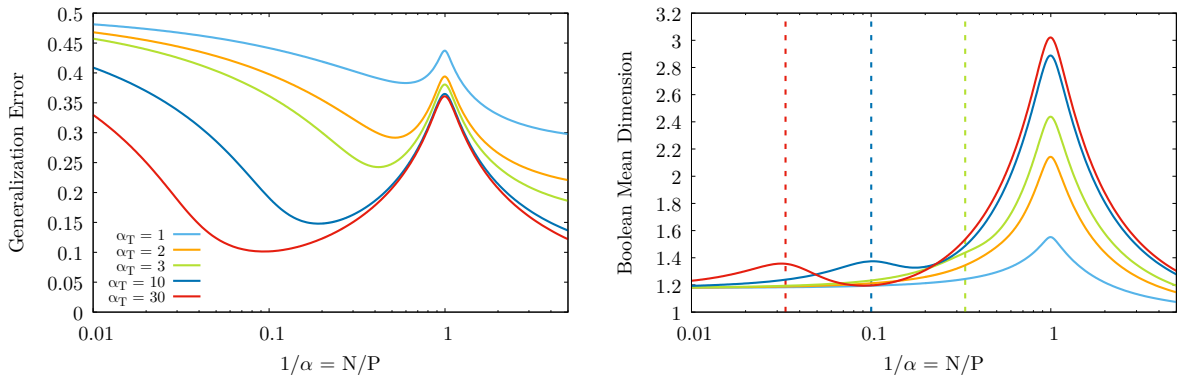

Figure 10: Generalization error (left) and BMD (right) as a function of $1/\alpha$ for $\lambda = 10^{-4}$ and $\sigma = \tanh$, for several values of $\alpha_T = P/D$. The loss used is the MSE. For low values of α_T the BMD displays a double descent behavior as showed in the main text. Increasing α_T, contrary to generalization, the BMD shows a triple descent behavior: a secondary peak in the BMD appears at $\alpha = \alpha_T$, i.e. $N = D$ (dashed vertical lines).

We remark here that this behavior observed in the BMD is reminescent of the triple descent behaviour observed in [d'Ascoli et al., 2020], but is nonetheless different in nature. Indeed, in [d'Ascoli et al., 2020] the triple descent was observed in the test loss, when fixing N and D (i.e. $\alpha_D = D/N$) and changing P[1]; the authors observe a peak in the test loss when $P = D$ in addition to the "classical" double descent peak when $P = N$. This "secondary" peak can be observed only if the activation function is linear $\sigma(x) = x$ or if the labels are corrupted by Gaussian noise $\zeta^\mu \sim \mathcal{N}(0, 1)$

$$y^\mu = \text{sign}\left(\frac{1}{\sqrt{D}} \sum_k w_k^T \xi_k^\mu + \sqrt{\Delta}\zeta^\mu \right) . \tag{82}$$

where Δ is a non-negative parameter modulating the noise intensity. It is easy to show that the only term to change in the free energy (64) because of the noise is the energetic term which is modified as

$$\mathcal{G}_E = 2 \int Dz_0 \, H\left(-\frac{M z_0}{\sqrt{Q_d - M^2 + \Delta}} \right) \max_{z_1} \left[-\frac{z_1^2}{2} - \ell(\sqrt{Q_d} z_0 + \sqrt{\delta Q} z_1) \right] . \tag{83}$$

[1] this is different from our setting where we fix P and D i.e. $\alpha_T = P/D$ and change N.

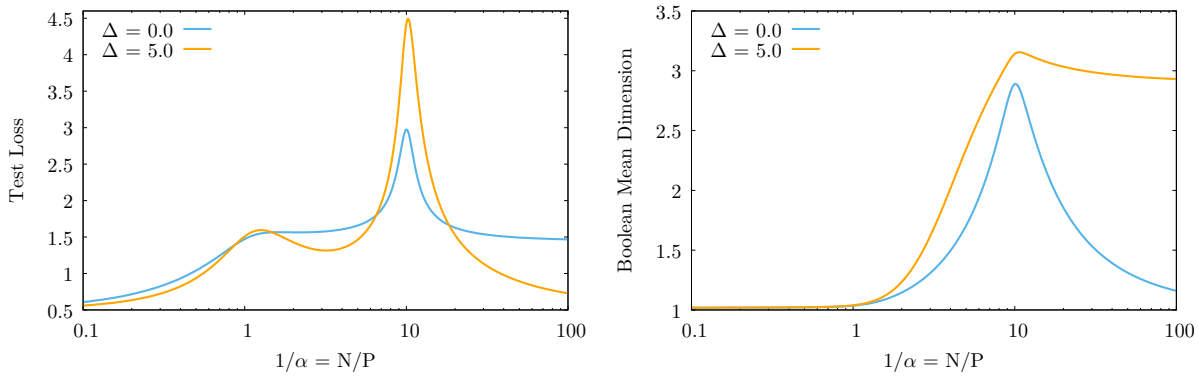

Figure 11: Test loss (left) and BMD (right) as a function of $1/\alpha$ for fixed $\alpha_D = 0.1$, $\lambda = 10^{-4}$, $\sigma = \tanh$ and for 2 values of the noise in the labels $\Delta = 0, 5$. The loss used is the MSE. Even if the test loss displays a secondary peak at $P = D$, the BMD does not.

We show in Fig. 11 that even if the test loss has a secondary peak when $\Delta > 0$ at $P = D$, this peak is not present in the BMD.

In Fig. 12, we show how the regularization λ can not only attenuate the "primary" peak at $P = N$, but also make the secondary peak disappear at $N = D$.

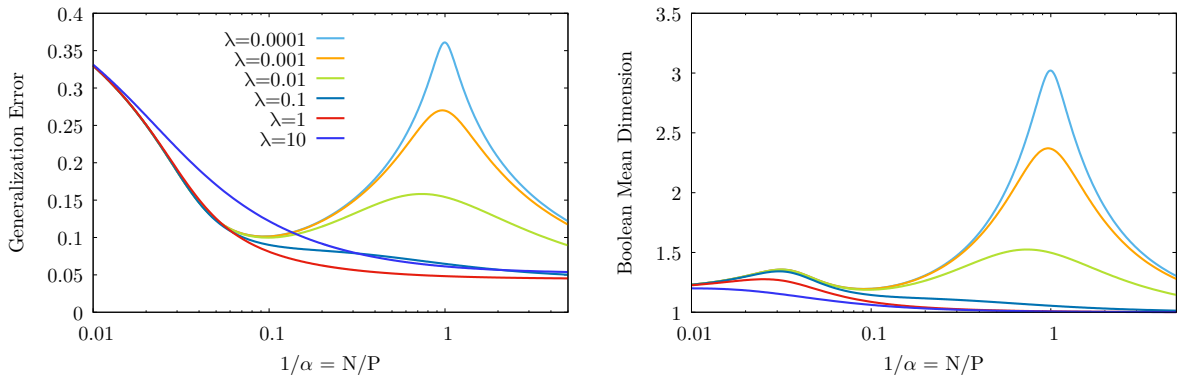

Figure 12: Generalization error (left) and BMD (right) as a function of $1/\alpha$ for $\alpha_T = 30$ and $\sigma = \tanh$ for several values of the regularization λ. The loss used is the MSE. Increasing the regularization the peak corresponding to $P = N$ disappears before the one located at $N = D$.

C.1 Explanation the secondary peak of the BMD around $N = D$

The root of this phenomenon can be found in the behavior of the spectrum of the covariance matrix $\Omega = F^T F/D$. To see this, we first consider a simplified setting where the phenomenon can be easily analytically traced. Consider a linear regression in the RFM with linear teacher. Calling $X_p \in \mathbb{R}^{P\times N} = \sigma(XF/\sqrt{D})$ the projected inputs of the RFM, with $X \in \mathbb{R}^{P\times D}$, and $F \in \mathbb{R}^{D\times N}$, the Gaussian Equivalence implies that the learning problem is equivalent to a linear regression with data:

$$X_p^{GET} = \kappa_1 XF/\sqrt{D} + \kappa_\star Z \tag{84}$$

with $Z \in \mathrm{R}^{P\times N}$ and $Z_{ij} \sim \mathcal{N}(0,1)$. The labels $Y \in \mathbb{R}^P$ are given by a linear teacher $w_T \in \mathbb{R}^D$:

$$Y = Xw_T \tag{85}$$

The ordinary least square (OLS) estimator gives a closed form solution for the trained weights:

$$\begin{aligned} w &= (X_p^T X_p/N + \lambda \mathbb{I})^{-1} X_p^T Y/\sqrt{N} \\ &= \frac{\alpha}{P}\left(\frac{\alpha}{P}(\kappa_1 \frac{XF}{\sqrt{D}} + \kappa_\star Z)^T(\kappa_1 \frac{XF}{\sqrt{D}} + \kappa_\star Z) + \lambda \mathbb{I}\right)^{-1}\left(\kappa_1 \frac{XF}{\sqrt{D}} + \kappa_\star Z\right)^T X w_T \\ &= \frac{\alpha}{P}\left(\frac{\alpha}{P}(\kappa_1 \frac{XF}{\sqrt{D}} + \kappa_\star Z)^T(\kappa_1 \frac{XF}{\sqrt{D}} + \kappa_\star Z) + \lambda \mathbb{I}\right)^{-1}\left(\kappa_1 \frac{XF}{\sqrt{D}} + \kappa_\star Z\right)^T X w_T \\ &= \frac{\alpha}{P}\left(\frac{\alpha}{P}\left(\kappa_1^2 \frac{F^T X^T X F}{D} + \kappa_\star^2 Z^T Z + \kappa_1 \kappa_\star \left(\frac{Z^T X F + F^T X Z}{\sqrt{D}}\right)\right) + \lambda \mathbb{I}\right)^{-1} \\ &\times \left(\kappa_1 \frac{F^T X^T X w_T}{\sqrt{D}} + \kappa_\star Z^T X w_T\right) \end{aligned} \tag{86}$$

By squaring this expression we can get the norm $q_d\, N = \|w\|^2$ of the OLS estimator. We would like to understand the behavior of this quantity when $P/N \to \infty$ and when $D/N = \alpha_D = \mathcal{O}(1)$ is varied. We can thus perform an annealed average over the dataset, averaging out X, Z, and w_T. Since we are going to square the expression, we can take advantage of:

$$\mathbb{E}\frac{X^T X}{P} = \mathbb{E}\frac{Z^T Z}{P} = \mathbb{I} \tag{87}$$

and defining $\Omega = F^T F/D$ we can simplify the expression as:

$$\begin{aligned} \mathbb{E}\|w\|^2 = \left(\kappa_1 \frac{w_T^T F}{\sqrt{D}} + \alpha \kappa_\star w_T^T \frac{X^T Z}{P}\right)\left(\left(\alpha\left(\kappa_1^2 \Omega + \kappa_\star^2 \mathbb{I}\right) + \lambda \mathbb{I}\right)^{-1}\right)^T \times \\ \times \left(\alpha\left(\kappa_1^2 \Omega + \kappa_\star^2 \mathbb{I}\right) + \lambda \mathbb{I}\right)^{-1}\left(\kappa_1 \frac{F^T w_T}{\sqrt{D}} + \alpha \kappa_\star \frac{Z^T X}{P} w_T\right) \end{aligned} \tag{88}$$

If we now move to the eigenbasis of $\Omega = V \Lambda X^T$, where we also have that $F/\sqrt{D} = U\sqrt{\Lambda} V^T$, we can write this expression as a trace over the eigenvalues in Λ:

$$q_d = \mathbb{E}\frac{\|w\|^2}{N} = \frac{\sigma_{w_T}^2}{N} \sum_{i=1}^{N} \frac{(\kappa_1^2 \rho_i + \alpha \kappa_\star^2)}{\left((\kappa_1^2 \rho_i + \kappa_\star^2) + \lambda\right)^2} \tag{89}$$

Similarly, one can get an expression for the average overlap:

$$Q_d = \mathbb{E}\frac{w^T(\kappa_1^2 \Omega + \alpha \kappa_\star^2) w}{N} = \frac{\sigma_{w_T}^2}{N} \sum_{i=1}^{N} \frac{(\kappa_1^2 \rho_i + \kappa_\star^2)^2}{\left((\kappa_1^2 \rho_i + \kappa_\star^2) + \lambda\right)^2} \tag{90}$$

So if we now focus on the ratio q_d/Q_d, which determines the fluctuations of the MD above $MD = 1$, we can see the impact of the spectrum of Ω, which follows a Marchenko-Pastur law with parameter $1/\alpha_D$. When $\alpha_D > 1$ the spectrum is continuous and strictly positive, with a minimum eigenvalue $\rho_- = (1 - \sqrt{1/\alpha_D})^2$. At $\alpha_D = 1$ the spectrum touches the origin, and then at smaller values of α_D (in the overparameterized regime of the RFM) the distribution splits into a delta in 0 with weight $1 - \alpha_D$ and a continuous component with increasing left extremum $\rho_- = (1 - \sqrt{1/\alpha_D})^2$ and weight α_D. Because of the additional ρ_i in the numerator of expression (90), when the eigenvalues of Ω approach zero they have a larger effect in q_d, therefore the MD reaches a peak.

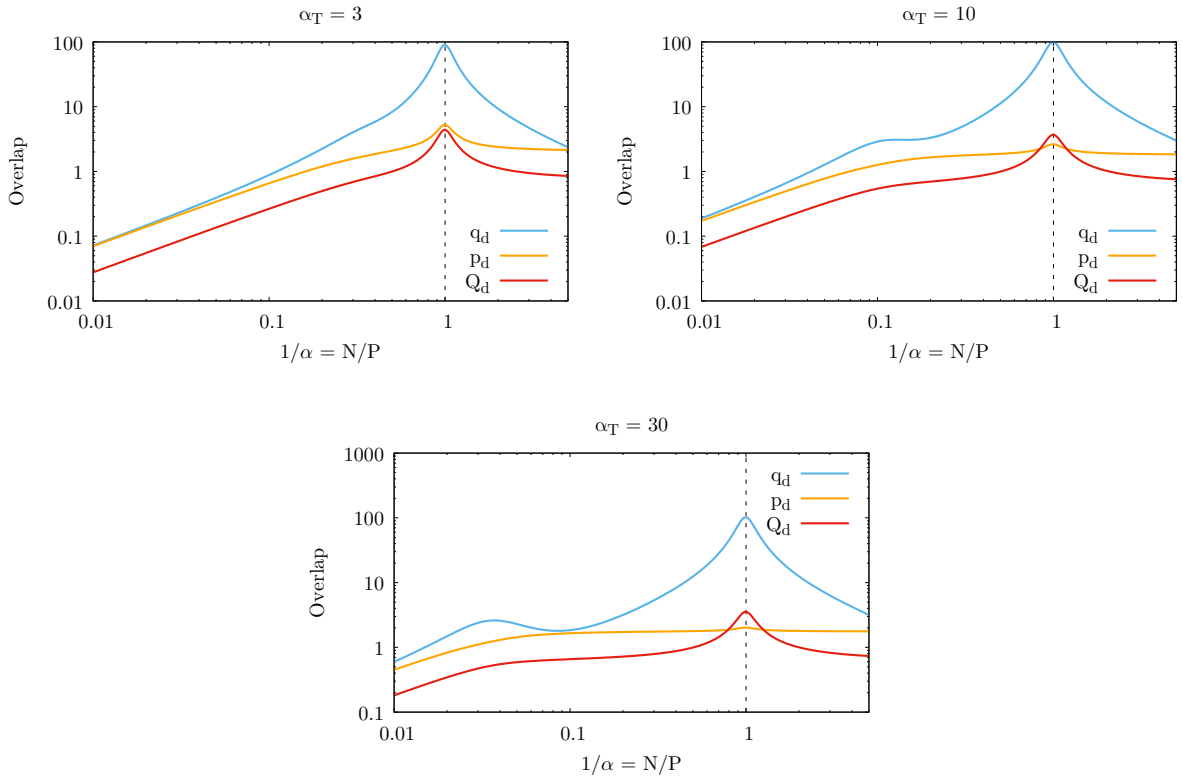

Figure 13: Plots of the overlaps q_d, p_d and $Q_d = \kappa_1^2 p_d + \kappa_\star^2 q_d$ as a function of $1/\alpha$ for $\alpha_T = 3, 10, 30$. In all plots the loss used is MSE, the regularization is $\lambda = 10^{-4}$ and $\sigma = \tanh$.

The same relationship between the parameters holds also at finite α and for a generic loss. The corresponding saddle-point equations read:

$$q_d = \frac{1}{N}\sum_{i=1}^{N} \frac{((\hat{m}^2 + \hat{q})\kappa_1^2\rho_i + \hat{q}\kappa_\star^2)}{\left(\delta\hat{q}(\kappa_1^2\rho_i + \kappa_\star^2) + \lambda\right)^2} \tag{91}$$

$$Q_d = \frac{1}{N}\sum_{i=1}^{N} \frac{((\hat{m}^2 + \hat{q})\kappa_1^2\rho_i + \hat{q}\kappa_\star^2)(\kappa_1^2\rho_i + \kappa_\star^2)}{\left(\delta\hat{q}(\kappa_1^2\rho_i + \kappa_\star^2) + \lambda\right)^2} \tag{92}$$

where the loss function and the number of constraints determine the value of the conjugate parameters $\hat{m}, \hat{q}, \delta\hat{q}$, but the presence of an additional $(\kappa_1^2\rho_i + \kappa_\star^2)$ in the numerator of Q_d induces the same behavior of the MD around $\alpha_D = D/N = 1$, independent of the specific setting.

In Fig. 13 we show the plot of the overlaps q_d, p_d and Q_d that confirms the intuitions showed above.

D Self-averaging property of the MD

The MD of the RFM at initialization demonstrates the self-averaging properties, e.g. for the case of i.i.d. gaussian weights that were projected on the space orthogonal to the $\mathbb{I}_D$ vector.

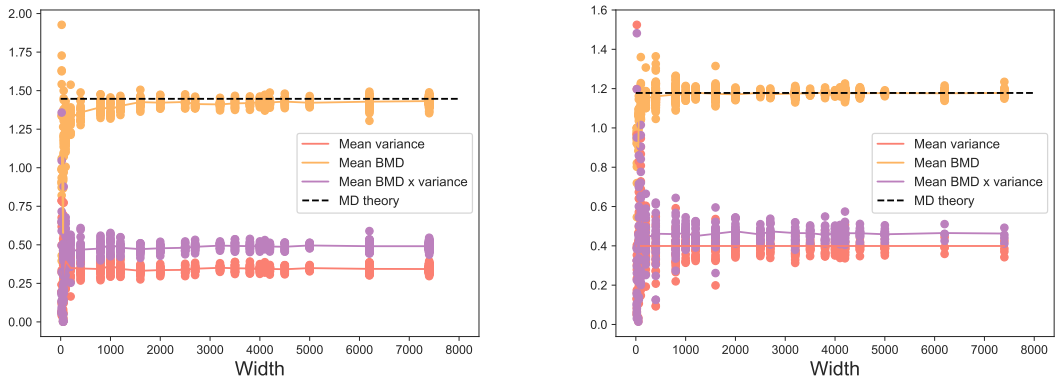

Figure 14: Correspondence of the empirical evidence of the self-averaging of the BMD computed for the RFM initialized with the gaussian weights (projected orthogonally to the $\mathbb{I}_D$ vector) and theoretical prediction of the asymptotic BMD value in the limit of the input size D and the hidden layer size N. (Left) RFM with leaky ReLU activation, (Right) RFM with tanh activation. The resulting plot represents an average over 70 different runs of the experiment.

D.1 Self-averaging of the MD for the trained models

In the Table 1 below we can observe the phenomenon of the concentration of the MD in the trained deep learning models as compared to the MD of the models at initialization.

Model	Var(MD) trained model	Var(MD) untrained model
ResNet20	0.02503	2.78465
ResNet32	0.0393	21.49987
ResNet44	0.05377	138.04827
ResNet56	0.0449	792.22086
mobilenetv2 x0 5	0.07485	318.17157
mobilenetv2 x0 75	0.05270	271.84772
mobilenetv2 x1 0	0.04614	81.66546
mobilenetv2 x1 4	0.03148	128.32706
shufflenetv2 x0 5	0.02638	146.33676
shufflenetv2 x1 0	0.0266	34.79431
shufflenetv2 x1 5	0.02843	27.3121
shufflenetv2 x2 0	0.03713	24.21249
repvgg a0	0.03894	104.79538
repvgg a1	0.04875	109.59382
repvgg a2	0.05133	136.44133
repvgg a0	3.44696	74.58045
repvgg a1	4.27395	83.92942
repvgg a2	2.66022	114.86224

Table 1: MD empirical variances for randomly initialized and trained on cifar-10 dataset deep models estimated over 30 seeds

E BMD and Data Normalization

In the Fig. 15, we repeat the BMD calculations for RFMs trained with different ranges used for normalization. As can be seen in the figure the choice of the input data normalization does not

affect the BMD pattern, but only its absolute values in this setting.

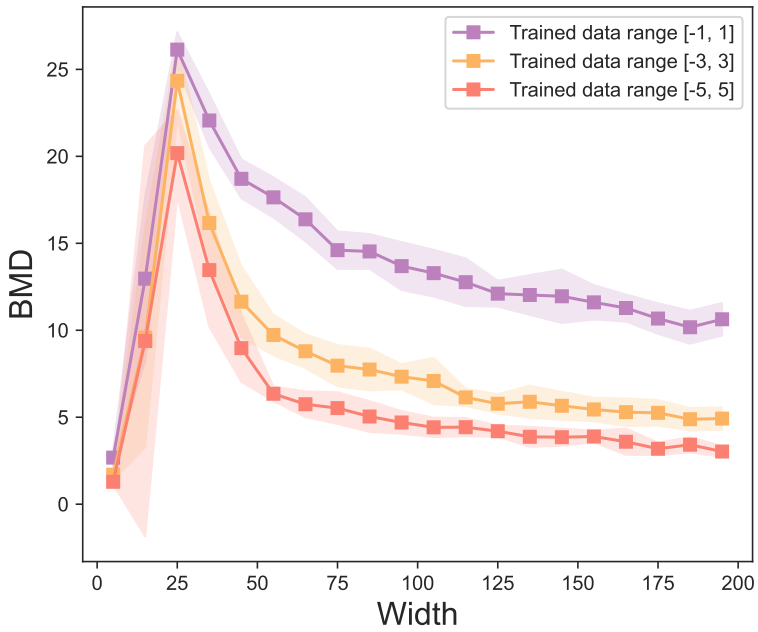

Figure 15: BMDs of the three RFMs trained on the MNIST dataset with 10 classes on 200 train samples with 0% label noise. The RFMs were trained on the datapoints normalized to lie within the intervals [-5, 5] (red line), [-3, 3] (orange line) and [-1, 1] (violet line). The resulting plot represents an average (and standard deviations) obtained repeating 12 different times the experiment.

F Effect of the label corruption on the train error

In the Fig. 16 we demonstrate the effect of the label corruption of the train data on the train error, which is allowing to explain the difference between the test and train errors for smaller model widths.

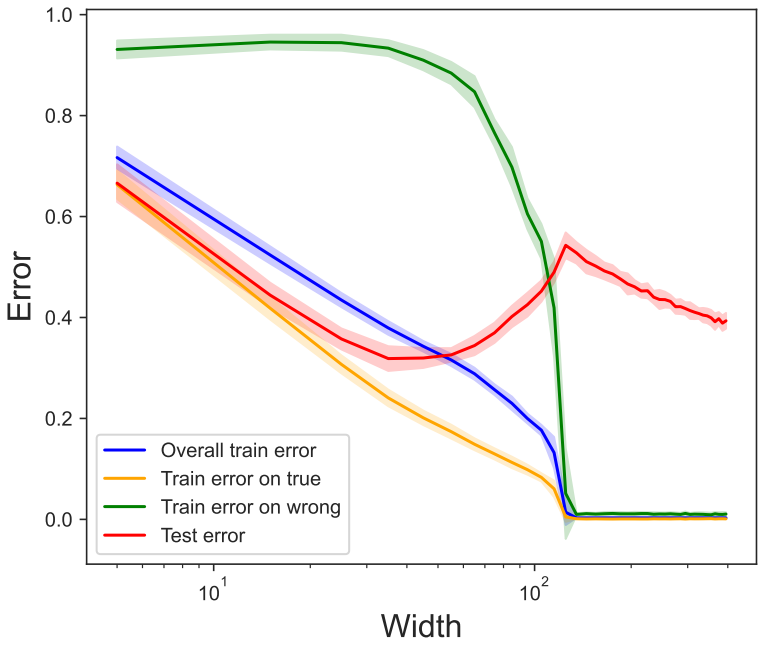

Figure 16: Train error curves of the RFM trained on the MNIST dataset with 10 classes on 1000 train samples with 20 % label noise and estimated on 1000 test samples. The plot demonstrates that better performance of the model on the test data rather than on the train data for smaller widths can be explained by a disproportionally high error on the train examples with the corrupted (wrong) labels (green line), which therefore leads to the higher overall train error (blue line), while tested only on the data with non-corrupted labels the train error (yellow line) is comparable to the test error (red line)