Proceedings of Machine Learning Research vol 145:1–46, 2021 2nd Annual Conference on Mathematical and Scientific Machine Learning

The Gaussian equivalence of generative models for learning with shallow neural networks

Sebastian Goldt SEBASTIAN.GOLDT@SISSA.IT
International School of Advanced Studies (SISSA), Trieste, Italy
Bruno Loureiro BRUNO.LOUREIRO@EPFL.CH
IdePHICS lab. Ecole Fédérale Polytechnique de Lausanne
Galen Reeves GALEN.REEVES@DUKE.EDU
Department of ECE and Department of Statistical Science, Duke University
Florent Krzakala FLORENT.KRZAKALA@EPFL.CH
IdePHICS lab. Ecole Fédérale Polytechnique de Lausanne
Marc Mézard MARC.MEZARD@ENS.FR
Laboratoire de Physique de l'Ecole Normale Supérieure, Université PSL, CNRS, Sorbonne Université, Université Paris-Diderot, Sorbonne Paris Cité, Paris, France
Lenka Zdeborová LENKA.ZDEBOROVA@EPFL.CH
SPOC lab. Ecole Fédérale Polytechnique de Lausanne

Editors: Joan Bruna, Jan S Hesthaven, Lenka Zdeborova

Abstract

Understanding the impact of data structure on the computational tractability of learning is a key challenge for the theory of neural networks. Many theoretical works do not explicitly model training data, or assume that inputs are drawn component-wise independently from some simple probability distribution. Here, we go beyond this simple paradigm by studying the performance of neural networks trained on data drawn from *pre-trained generative models*. This is possible due to a Gaussian equivalence stating that the key metrics of interest, such as the training and test errors, can be fully captured by an appropriately chosen Gaussian model. We provide three strands of rigorous, analytical and numerical evidence corroborating this equivalence. First, we establish rigorous conditions for the Gaussian equivalence to hold in the case of single-layer generative models, as well as deterministic rates for convergence in distribution. Second, we leverage this equivalence to derive a closed set of equations describing the generalisation performance of two widely studied machine learning problems: two-layer neural networks trained using one-pass stochastic gradient descent, and full-batch pre-learned features or kernel methods. Finally, we perform experiments demonstrating how our theory applies to deep, pre-trained generative models. These results open a viable path to the theoretical study of machine learning models with realistic data.

Keywords: Neural networks, Generative models, Stochastic Gradient Descent, Random Features.

1. Introduction

Consider a supervised learning task where we are given a stream of samples drawn i.i.d. from an unknown distribution $q(x, y)$. Each sample consists of an input vector $x = (x_i) \in \mathbb{R}^N$ and a response or label $y \in \mathbb{R}$. Our goal is to learn a function $\phi_\theta : \mathbb{R}^N \to \mathbb{R}$ with parameters θ that provides an estimate of y given x. The performance of such a model ϕ_θ at this task is assessed in terms of its prediction or test error $\mathsf{pe}(\theta) = \mathbb{E}\,\ell\left[y, \phi_\theta(x)\right]$, where the expectation is over the data distribution $q(x, y)$ for a fixed set of parameters θ and some loss function ℓ. A lot of attention has

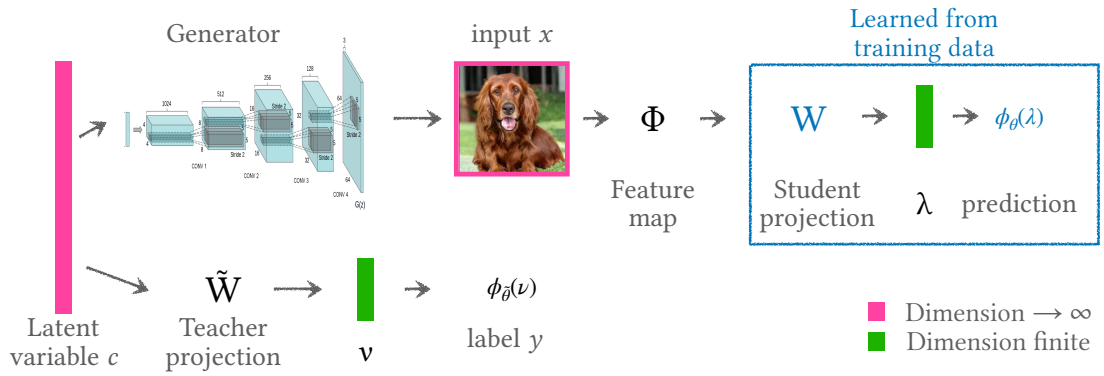

Figure 1: **The deep hidden manifold: going beyond the i.i.d. paradigm for generating data in the teacher-student setup.** We analyse a setup where samples (x, y) are generated by first drawing a latent vector $c \sim \mathcal{N}(0, I_D)$. The input x is obtained by propagating the latent vector through a (possibly deep) generative network, $x = \mathcal{G}(c)$. The label y is given by the response of a two-layer teacher network to the latent vector. We then analyse in a closed form learning via a two-layer neural network, or with a single layer neural network after a projection trough a fixed, but not necessarily random, feature map. The sketch of the generator is taken from Radford et al. (2016), whose deep convolutional GAN is one of the generators we use in our experiments in Sec. 4.

recently focused on the importance of training to find models ϕ_θ with low test error, and specifically on the role of stochastic gradient descent and various regularisations. Analysing the impact of the data distribution $q(x, y)$ on learning is equally important, yet it is not well understood.

In fact, theoretical works on learning in statistics or theoretical computer science traditionally try to make only minimal assumptions on the class of distributions $q(x, y)$ Mohri et al. (2012); Vapnik (2013) or consider the case where data are chosen in an adversarial (worst-case) manner. In a complementary line of work that emanated originally from statistical physics Gardner and Derrida (1989); Seung et al. (1992); Watkin et al. (1993); Engel and Van den Broeck (2001); Zdeborová and Krzakala (2016), inputs are modelled as high-dimensional vectors whose elements are drawn i.i.d. from some probability distribution. Their labels are either assumed to be random, or given by some random, but fixed function of the inputs, see Fig. 1 (a). This approach, known as the teacher-student setup, has recently experienced a surge of activity in the machine learning community Zhong et al. (2017); Tian (2017); Du et al. (2018); Soltanolkotabi et al. (2018); Aubin et al. (2018); Saxe et al. (2018); Baity-Jesi et al. (2018); Goldt et al. (2019); Ghorbani et al. (2019); Yoshida and Okada (2019); Gabrié (2020); Bahri et al. (2020); Zdeborová (2020); Advani et al. (2020)

The deep hidden manifold In this manuscript we go beyond the i.i.d. paradigm of the teacher-student setup by extending the hidden manifold model analysed in Goldt et al. (2020); Gerace et al. (2020). Fig. 1 gives a visual overview of the components of the model. We draw the inputs x from a generative model $\mathcal{G} : \mathbb{R}^D \to \mathbb{R}^N$ of depth L. These models transform random uncorrelated latent variables $c = (c_r) \in \mathbb{R}^D$ into correlated, high-dimensional inputs which follow a given target

distribution via

$$x = \mathcal{G}(c) = \mathcal{G}^L \cdots \mathcal{G}^3 \circ \mathcal{G}^2 \circ \mathcal{G}^1(c), \qquad c \sim \mathcal{N}(0, I_D), \tag{1}$$

where $\circ$ denotes the chaining of layers $\mathcal{G}^\ell$, which could be fully-connected, convolutional Fukushima and Miyake (1982); LeCun et al. (1990), applying batch norm Ioffe and Szegedy (2015) or an invertible mapping $\mathcal{G}^\ell : \mathcal{R}^D \to \mathcal{R}^D$ as they are used normalising flows. We thus replace i.i.d. Gaussian inputs with realistic images such as the one shown in Fig. 1. While Goldt et al. (2020); Gerace et al. (2020) only studied generative models with a single layer of weights, we allow the generator to be of arbitrary depth L, thus including important models such as variational auto-encoders Kingma and Welling (2014), generative adversarial networks (GAN) Goodfellow et al. (2014), or normalising flows Tabak et al. (2010); Tabak and Turner (2013); Rezende and Mohamed (2015).

The label for each input is obtained from a two-layer teacher network with M hidden neurons and parameters $\tilde{\theta} = (\tilde{v} \in \mathbb{R}^M, \tilde{W} \in \mathbb{R}^{M \times D})$ acting on the *latent representation* c of the input,

$$y = \sum_{m=1}^{M} \tilde{v}^m \, \tilde{g}\left(\nu^m\right), \quad \nu^m \equiv \frac{1}{\sqrt{D}} \sum_{r=1}^{D} \tilde{w}_r^m c_r. \tag{2}$$

The intuition here comes from image classification, where the label of an image does not depend on every pixel x, but the higher-level features of the image, which should be better captured by its lower-dimensional latent representation, like in conditional generative models Mirza and Osindero (2014); Brock et al. (2019). We call this the *deep hidden manifold model*.

The two models of learning that we analyse The advantage of the vanilla teacher-student setup is that it lends itself well to analytical studies, at the detriment of having unrealistic inputs. The deep hidden manifold allows us to study realistic inputs, but can we still analyse it? We provide two distinct positive answers to this question for two common parametric models $\hat{y} = \phi_\theta(x)$ trained on a dataset with i.i.d. samples $\mathcal{D}_T = \{(x^\mu, y^\mu)\}_{\mu=1}^T$ generated by the deep hidden manifold q. First, we provide a sharp asymptotic analysis of **full-batch learning with pre-learned features** Rahimi and Recht (2008):

$$\phi_\theta(x) = g\left(\lambda\right), \qquad \lambda = \frac{1}{\sqrt{\tilde{N}}} \hat{w}^\top \sigma\left(Fx\right), \tag{3}$$

where $F \in \mathbb{R}^{\tilde{N} \times N}$ defines the feature map $\Phi_{\mathrm{F}} = {}^{\sigma(F\,\cdot\,)}/_{\sqrt{N}} \colon \mathbb{R}^N \to \mathbb{R}^{\tilde{N}}$, which is not necessarily random. We obtain the weights $\hat{w} \in \mathbb{R}^{\tilde{N}}$ by minimising the empirical risk in feature space:

$$\hat{w}_T = \underset{w \in \mathbb{R}^{\tilde{N}}}{\operatorname{argmin}} \left[\sum_{\mu=1}^{T} \ell\left(y^\mu, w^\top \Phi_F(x^\mu)\right) + \frac{\lambda}{2} ||w||_2^2 \right] \tag{4}$$

with a convex loss function ℓ and a ridge penalty term $\lambda > 0$. In this model, the asymptotic limits is defined by taking $T, N, \tilde{N} \to \infty$ with fixed ratios $\tilde{N}/N, T/\tilde{N} \sim O(1)$.

Second, we provide an asymptotic analysis of **one-pass stochastic gradient descent in a two-layer neural network** with $K \sim O(1)$ hidden units:

$$\phi_\theta(x) = \sum_{k=1}^{K} v^k \, g\left(\lambda^k\right), \quad \lambda^k \equiv \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^k x_i, \tag{5}$$

where we take $N \to \infty$. In this case, the network is trained end-to-end with stochastic gradient descent on the quadratic loss using a previously unseem sample at each step μ of training:

$$\mathrm{d}w_i^k \equiv \left(w_i^k\right)_{\mu+1} - \left(w_i^k\right)_{\mu} = -\frac{\eta}{\sqrt{N}} v^k \Delta g'(\lambda^k) x_i, \qquad \mathrm{d}v^k = -\frac{\eta}{N} g(\lambda^k)\Delta. \tag{6}$$

where $\Delta = \sum_{j=1}^{K} v^j g(\lambda^j) - \sum_{m=1}^{M} \tilde{v}^m \tilde{g}(\nu^m)$. Note the different scaling of the learning rate η, which guarantees the existence of a well-defined limit of the SGD dynamics as $N \to \infty$.

Test error and Gaussian equivalence property In both cases, the learner is thus given a dataset $\mathcal{D}_T = \{(x^\mu, y^\mu)\}_{\mu=1}^{T}$ consisting of T i.i.d. samples from $q(x, y)$. The classifier ϕ_θ with parameters $\theta = (W, v)$ either acts directly on the inputs x or on a feature map $\Phi\colon \mathbb{R}^N \to \mathbb{R}^{\tilde{N}}$. The learning algorithm produces θ_T based on the training data. The model ϕ is evaluated using the prediction MSE, which for each θ is

$$\mathsf{pmse}(q, \theta) \equiv \frac{1}{2} \int_{\mathbb{R}^N \times \mathbb{R}} (\phi_\theta(\Phi(x)) - y)^2 \, \mathrm{d}q(x, y) \tag{7}$$

The key observation in our analysis is that for both models (3) and (5) and the teacher (2), the respective inputs only enter via the "pre-activations" $\lambda = (\lambda^k)$ and $\nu = (\nu^m)$. We can therefore replace the high-dimensional average over $q(x, y)$ by a low-dimensional average over the joint distribution $P_\theta(\lambda, \nu)$ of (λ, ν), which is a function of $\theta = (W, v)$:

$$\mathsf{pmse}(q, \theta) \equiv \frac{1}{2} \int_{\mathbb{R}^K \times \mathbb{R}^M} \left(\sum_{k=1}^{K} v^k \, g(\lambda^k) - \sum_{m}^{M} \tilde{v}^m \tilde{g}(\nu^m) \right)^2 \mathrm{d}P_\theta(\lambda, \nu) \tag{8}$$

The complexity of the high-dimensional distribution q is thus encapsulated by the low-dimensional distribution $P_\theta(\lambda, \nu)$. If the student weights W are drawn element-wise i.i.d. from some distribution irrespective of the training data and the (transformed) inputs of the student $\tilde{x}$ are weakly correlated on average, then (λ, ν) are jointly Gaussian with high probability over W if. Equivalently, we can require some spectral condition on the covariance matrix of $\tilde{x}$.

To be precise, consider a sequence of models and parameters (q, θ), where we let the dimension the latent space D, the dimension of the data N, and the dimension of the features $\tilde{N}$ scale to infinity at the same rate, while keeping the dimensions of (λ, ν) fixed. The **Gaussian equivalence property** (GEP) is said to hold if $P_\theta(\lambda, \nu)$ is asymptotically Gaussian, i.e., $d(P_\theta, P_\theta^*) = o_N(1)$ where P_θ^* is the Gaussian probability distribution with the same first and second moments and $d(\cdot, \cdot)$ is a metric that metrizes convergence in distribution and in second moments.

The Gaussian Equivalence property simplifies the analysis significantly, since it allows for the pmse to be evaluated asymptotically in terms of the finite dimensional Gaussian integral

$$\mathsf{pmse}\,(q, \theta) \to \frac{1}{2} \int_{\mathbb{R}^K \times \mathbb{R}^M} \left(\sum_{k=1}^{K} v^k \, g(\lambda^k) - \sum_{m}^{M} \tilde{v}^m \tilde{g}(\nu^m) \right)^2 \mathrm{d}P_\theta^*(\lambda, \nu). \tag{9}$$

The pmse is thus a function of only the second moments of (λ, ν):

$$Q^{k\ell} \equiv \mathbb{E}\, \lambda^k \lambda^\ell, \qquad R^{km} \equiv \mathbb{E}\, \lambda^k \nu^m, \qquad T^{mn} \equiv \mathbb{E}\, \nu^m \nu^n, \tag{10}$$

and of the second-layer weights v^k and $\tilde{v}^m$ in the case of two-layer neural networks. This reduction of the high-dimensional average (7) to an expression in terms of an $O(1)$ number of "order parameters" is central to the vast literature analysing the vanilla teacher-student setup Gardner and Derrida (1989); Seung et al. (1992); Watkin et al. (1993); Biehl and Schwarze (1995); Saad and Solla (1995a); Engel and Van den Broeck (2001).

Surprisingly, here we find that this reduction also holds if the weights of the student are obtained from the training data using the algorithms (6) and (4). Hence, despite the correlations of the weights to the correlated inputs, a characterisation of the pmse for models like Eq. (3) and (5) in terms of scalar order parameters remain true for many generative data models, including common trained deep generative networks, *during* learning. This observation can be formalised in the following conjecture, which is the central claim of our paper:

Conjecture 1 (Deep Gaussian Equivalence Conjecture) *Suppose that 1) the teacher weights $\tilde{W}$ are generated i.i.d. and 2) $\mathsf{Cov}(\Phi(x))$ satisfies some weak correlation property. Let $\hat{\theta}_T$ be obtained from either online SGD (6) or empirical risk minimisation (4). Then, the GEP holds in the sense that for some probability distance $d(\cdot,\cdot)$, we have*

$$d\left(P_{\theta_T}, P^*_{\theta_T}\right) \to 0 \quad \textit{in probability} \tag{11}$$

as $N, D, T \to \infty$ with $N, T = \Theta(D)$ and $M, K = O(1)$. Here, the probability is taken with respect to the randomness q (i.e, the teacher weights and any other random components in the generator), the feature map Φ, which may or may not be random, and the training data $\mathcal{D}_T$.

We believe it is an exciting research direction to establish the limits of Conjecture 1. In this manuscript we give the first steps in this direction by presenting three strands of rigorous (Sec. 2), analytical (Sec. 3) and numerical (Sec. 4) evidence that the conjectured "deep GEC" holds true for different tasks on shallow networks and for a wide range of deep, pre-trained generative models. In particular, we provide:

(i) A rigorous proof of Conjecture 1 for a single-layer generator of the form $\mathcal{G}(c) = \sigma(Ac)$ where A is a matrix with pre-trained weights, and σ is a point-wise non-linearity. Our **Gaussian equivalence theorem** (GET, Thm. 2) gives sufficient conditions on the weights A under which a given low-dimensional projection of the input x, such as λ, ν, is approximately Gaussian. We thus put the Gaussian equivalence property used in Goldt et al. (2020); Gerace et al. (2020) on a rigorous basis.

(ii) An exact analytical description of the evolution of the test error of a two-layer neural network trained using one-pass (or online) SGD (Sec. 3.1), whose predictions exactly match simulations with convolutional GANs and normalising flows pre-trained on CIFAR10 (Sec. 4).

(iii) A set of scalar self-consistent equations describing the test error for full-batch learning of T i.i.d. samples using regression with $\tilde{N}$ features in the regime where $N, \tilde{N}, D, T \to \infty$ with $T/\tilde{N}, N/\tilde{N} = O(1)$ (Sec. 3.2). As before, we confirm the accuracy of this theoretical prediction with experiments of convolutional GANs pre-trained on CIFAR100 (Sec. 4).

Further related work Several works have recognised the importance of data structure in machine learning, and in particular the need to go beyond the simple component-wise i.i.d. modelling for neural networks Bruna and Mallat (2013); Patel et al. (2016); Mossel (2016); Gabrié et al. (2018), recurrent neural networks Mézard (2017) and inference problems such as matrix factorisation Hand

et al. (2018); Aubin et al. (2019). Ansuini et al. (2019) demonstrated that a network's ability to transform data into low-dimensional manifolds was predictive of its classification accuracy.

While we will focus on the prediction error, a few recent papers studied a network's ability to *store* inputs with lower-dimensional structure and random labels: Chung et al. (2018b) studied the linear separability of general, finite-dimensional manifolds and their interesting consequences for the training of deep neural networks Chung et al. (2018a); Cohen et al. (2020), while Cover's classic argument Cover (1965) to count the number of learnable dichotomies was recently extended to cover the case where inputs are grouped in tuples of k inputs with the same label Rotondo et al. (2020); Borra et al. (2019). Koehler and Risteski (2019) studied the expressive power of ReLU networks compared to polynomial kernels under a data model where the teacher is a linear function of c, and the inputs are a noisy linear projection of the latent variables. Recently Yoshida and Okada (2019) analysed the dynamics of online learning for data having an arbitrary covariance matrix, finding an infinite hierarchy of ODEs (cf. Sec. 3.1).

Gaussian equivalent models are currently attracting a lot of interest. During the revision of this work, we became aware of a recent alternative proof of the GET by Hu and Lu (2020) for a slightly different setup. A parallel line research analysed random features regression using random matrix theory (RMT) Louart et al. (2018); Fan and Montanari (2019). The equivalent mapping to a Gaussian model with appropriately chosen covariance was explicitly stated and used in Mei and Montanari (2019); Montanari et al. (2019) and extended to a broader setting encompassing data coming from a GAN in Seddik et al. (2019, 2020). We will discuss these works in relation to our results in Sec. 2.2.

Reproducibility We provide code to solve the equations of Sec. 3 and the experiments of Sec. 4 online at `https://github.com/sgoldt/gaussian-equiv-2layer`.

2. The Gaussian Equivalence Theorem

We start with the study of a simple generator where inputs are generated according to

$$x_n = \mathcal{G}_n(c) = \sigma(a_n^\top c) = \sigma\left(\sum_{r=1}^{D} a_{rn} c_r\right) \tag{12}$$

where $N \to \infty$, $D \to \infty$ at fixed $\delta = D/N$, and $\sigma : \mathbb{R} \to \mathbb{R}$ is a non-linear function and $A = [a_1, \dots, a_n]^\top$ is the weight matrix of the generator. This is precisely the setting of the hidden manifold model of Goldt et al. (2020); Gerace et al. (2020), and generators of the form (12) cover a number of important cases beyond the hidden manifold model: *(i)* random feature models Rahimi and Recht (2008, 2009), which regard the latent variable c as the true underlying data and x as features constructed from c that are used as inputs for the prediction algorithm (cf. Sec. 3.2); *(ii)* Gaussian feature models, where the inputs x are jointly Gaussian with the latent variables c; and *(iii)* the classic teacher-student setup Gardner and Derrida (1989); Seung et al. (1992); Engel and Van den Broeck (2001), where the features x are equal to the latent variables c.

The inputs generated by such a generator are not Gaussian. However, our first main result, the **Gaussian Equivalence Theorem**, guarantees that the local fields (λ, ν) are still jointly Gaussian, and hence a description in terms of order parameters like Eq. (9) possible, even if inputs are drawn from this generator. More precisely, the theorem gives verifiable conditions on σ and the weight matrices of the student, teacher and generator networks, under which a low-dimensional projection of the inputs, such as λ and ν, is approximately Gaussian.

2.1. Statement of the theorem

Given probability measures P and Q on $\mathbb{R}$, define

$$d(P,Q) \equiv \sup_{f\in\mathcal{F}} \left|\mathbb{E}_P[f] - \mathbb{E}_Q[f]\right|, \tag{13}$$

where $\mathcal{F} = \{f \,:\, \|f''\|_\infty, \|f'''\|_\infty \leq 1\}$ is the set of thrice-differentiable functions with bounded second and third derivative and $\|f\|_\infty$ is the uniform norm of f. Given probability measures P and Q on $\mathbb{R}^d$ the maximum-sliced (MS) distance is defined by

$$d_{\mathrm{MS}}(P,Q) \equiv \sup_{\alpha\,:\,\|\alpha\|\leq 1} d(\alpha^\top P, \alpha^\top Q) \tag{14}$$

where $\alpha^\top P$ denotes the one-dimensional distribution corresponding to the projection of P into the direction of α. It can be verified that the MS distance is a metric Kolouri et al. (2019) and that convergence with respect to d_{MS} implies convergence in distribution as well as convergence of second moments. Our result requires the following regularity assumptions:

A1) Row normalisation $\|a_n\| = 1$;

A2) Smoothness: the non-linearity σ is thrice differential with $\mathbb{E}\left[|\sigma(u)|^4\right]$, $\mathbb{E}\left[|\sigma'(u)|^2\right]$, and $\mathbb{E}\left[|\sigma''(u)|^2\right]$ all $O(1)$ for $u \sim \mathcal{N}(0,1)$;

A3) Bounded student weights: $w_n^k = O(1)$.

Note that the smoothness assumption on the non-linearity σ can be relaxed to the assumption that σ is Lipschitz continuous, with the only consequence being a loss in the rate of convergence. The basic idea is that any Lipschitz function can be approximated by a function that satisfies the smoothness assumptions, see e.g. (O'Donnell, 2014, Proposition 11.58). The dependence on σ is quantified in terms the first, second, and third Hermite coefficients, which are defined by

$$\hat{\sigma}(1) \equiv \mathbb{E}\left[\sigma(u)u\right], \quad \hat{\sigma}(2) \equiv \frac{1}{\sqrt{2}}\mathbb{E}\left[\sigma(u)(u^2-1)\right], \quad \hat{\sigma}(3) \equiv \frac{1}{\sqrt{6}}\mathbb{E}\left[\sigma(u)(u^3-3u)\right], \tag{15}$$

where the expectation is taken with respect to a standard Gaussian random variable u. Furthermore, let $\rho = AA^\top$ and $\tilde{\rho} = \rho - I_N$ and define the $N \times N$ matrices:

$$M_1 = \frac{1}{\sqrt{N}}\left(\hat{\sigma}^2(1)\tilde{\rho}^2 + \hat{\sigma}^2(2)\tilde{\rho}^2 \circ \rho\right), \qquad M_2 = \hat{\sigma}^2(2)\left(\tilde{\rho}\circ\tilde{\rho}\right)^2 + \hat{\sigma}^2(3)\left(\tilde{\rho}\circ\tilde{\rho}\right)^2 \circ \rho, \tag{16}$$

where $\circ$ denotes the Hadamard entrywise product. Each of these matrices is positive semi-definite, by the Schur product theorem (Horn and Johnson, 2012, Sec. 7.5), and thus has a unique positive semi-definite square root. We then have:

Theorem 2 (Gaussian Equivalence Theorem) *Let P be the distribution of the pair (λ,ν) and let $\hat{P}$ be the Gaussian distribution with the same first and second moments. Under Assumptions A1-A3,*

$$d_{\mathrm{MS}}(P,\hat{P}) = O\left(\left\|\tfrac{1}{\sqrt{N}}WM_1^{1/2}\right\|^2 + \left\|\tfrac{1}{\sqrt{N}}WM_2^{1/2}\right\| + \tfrac{1}{\sqrt{N}}\left\|\tfrac{1}{\sqrt{D}}\tilde{W}A^\top\right\|^2 + \frac{1+\sum_{i\neq j}(a_i^\top a_j)^4}{\sqrt{N}}\right). \tag{17}$$

We provide the proof of Theorem 2 in Sec. A.

2.2. Discussion

Theorem 2 can be viewed as a multivariate central limit theorem (CLT) for weakly dependent random variables. The terms involving the matrices M_1 and M_2 quantify the impact of the dependencies in x. Note for example that if the columns of A are uncorrelated, then both of these terms are zero and Theorem 2 recovers a variation of the classical Berry–Esseen Theorem (O'Donnell, 2014, Chapter 11.5). The significance of Theorem 2 is that it provides a simple and verifiable sufficient condition for the joint Gaussianity of (λ, ν) for pre-trained, and hence correlated generator weights. The basic idea is that in order for Gaussianity to hold, the weight matrices should avoid any directions in the matrices M_1 and M_2 associated with eigenvalues that are not converging to zero.

To appreciate how the spectral properties of M_1 and M_2 depend on A and σ, it is useful to consider some examples. We give two quick examples below; we discuss these examples in detail in Sec. B, where we analyse how the leading eigenvalues and eigenvectors of M_1 and M_2 depend on A using analytical and numerical arguments.

Example 1 (IID A) *If the entries of A are i.i.d. sub-Gaussian, then $\|M_1\| = O(1/\sqrt{N})$ with high probability. If $\hat{\sigma}^2(2)$ is nonzero, then M_2 has one eigenvalue that is $O(1)$ associated with the all-ones vector and the rest are $O(1/N)$. If $\hat{\sigma}^2(2) = 0$, which occurs whenever σ is an odd function, then $\|M_2\| = O(1/N)$. Thus, if $\hat{\sigma}(2) = 0$ or $\|\frac{1}{\sqrt{N}} W\mathbf{1}\| = O(1/N)$ it follows that $d_{\mathrm{MS}}(P, \hat{P}) = O(1/\sqrt{N})$ with high-probability over A.*

Example 2 (Deterministic A) *Next consider the case $(D \geq N)$ where*

$$AA^\top = I_N + \frac{c}{\sqrt{N}}(\mathbf{1}_N - I_N)$$

for some fixed constant c. Suppose that $\sigma(k), k = 1, 2, 3$ are nonzero. Direct calculation reveals that M_1 has one eigenvalue $O(\sqrt{N})$ with the rest $O(1/\sqrt{N})$ and M_2 has one eigenvalue $O(1)$ with the rest $O(1/N)$. In both cases, the leading eigenvector is proportional to the all ones vector. Thus if $\|\frac{1}{\sqrt{N}} W\mathbf{1}\| = O(1/N)$ then $d_{\mathrm{MS}}(P, \hat{P}) = O(1/\sqrt{N})$.

The idea that most low-dimensional projections of a high-dimensional distribution are approximately random has a rich history Sudakov (1978); Diaconis and Freedman (1984); Hall and Li (1993); Bobkov (2003); Meckes (2010); Reeves (2017). In this line of work, "most" is quantified in terms of high-probability guarantees with respect to a random weight matrix W that is independent of x. For example, if the entries of W are i.i.d. standard Gaussian, then the necessary and sufficient conditions for convergence to a Gaussian are that 1) $1/n\|x\|^2$ concentrates about is mean 2) and $1/n\|\,\mathsf{Cov}(x)\|_F^2 \to 0$ (assuming zero mean). In the setting of this paper, it can be verified that these properties are implied by assumptions A1 and A2. The added benefit of Theorem 2 is that "most" is now quantified deterministically in terms of the number of the eigenvalues of M_1 and M_2.

The last term in (17) imposes a constraint on the *average* pairwise correlation between the columns of A. Specifically, this term converges to zero provide that $\sum_{i \neq j} (a_i^\top a_j)^4 = o(\sqrt{N})$. Importantly, this constraint still allows for allows for the possibility that a subset of the entries of A have correlation of order one. By contrast, previous work in this setting requires either randomly generated features or a much stronger incoherence constraint on the *maximum* correlation between any two entries. The generality provided by A1 is crucial to our target applications since it allows for

"sufficiently small" subsets to have arbitrary dependence structure. This is also a key difference to the proof of a similar result by Hu and Lu (2020) that appeared during the revision of this manuscript.

Our analysis also highlights the dependence of the first few terms in the Hermite expansion of σ. While Hu and Lu (2020) assume that σ is odd, which leads to $\hat{\sigma}(2) = 0$, our analysis highlights the crucial role of $\hat{\sigma}(2)$: if it is non-zero, as is the case for ReLU, then correlation in λ is described not by the linear dependence with ν, but by a quadratic dependence, leading to more stringent conditions for the validity of the CLT.

In a different direction, Gaussian behaviour associated with random choices of the parameter A have also been studied in the context of infinitely wide networks Neal (1995); Lee et al. (2018); de G. Matthews et al. (2018). Specifically, if the entries of A are i.i.d. Gaussian random variables it follows that $\lambda \mid \nu$ can be viewed as Gaussian processes indexed by ν. Combined with the Gaussianity of ν, this establishes the GET under general conditions on the generator. However, this analysis relies crucially on the assumption that A is generated independently of everything else. This assumption precludes the application to pre-trained generators.

A recent line research has derived Gaussian equivalence theorems for generators with random weights using random matrix theory (RMT) Hachem et al. (2007); Cheng and Singer (2013); Pennington and Worah (2017); Louart et al. (2018); Fan and Montanari (2019). The equivalent mapping to a Gaussian model with appropriately chosen covariance was explicitly stated and used in Mei and Montanari (2019); Montanari et al. (2019) and extended to a broader setting encompassing data coming from a GAN in Seddik et al. (2019, 2020). Similar to the analysis in this paper, the high-level idea is that certain integrals with respect to the data distribution $q(x, y)$ can be replaced by integrals over an appropriately defined Gaussian approximation. The main difference is the class of functions considered. Specifically, Theorem 2 provides guarantees for any sufficiently smooth function applied to a given low-dimensional projections of the features (x, c). This form of approximation is needed to justify the integro-differential equations derived in Sec. 3.1. By contrast, the RMT approach provides guarantees for a restricted set of functions applied to high-dimensional matrices derived from samples of (x, c). For example, these results provide equivalence of the empirical spectral measures of these random matrices as well as the test error associated with specific learning algorithms. The results in this paper thus neither imply previous work, nor are they, to the best of our knowledge, implied by it.

3. Analysis of neural networks learning on data from deep generators

We now turn to two applications of the deep GEC that allow us to analyse learning in paradigmatic model systems in detail, and at the same time help us gather experimental evidence for the deep GEC. We will first derive a set of equations that describe the evolution of the test error of a two-layer neural network trained using one-pass (or online) SGD on the deep hidden manifold model (Sec. 3.1). We also use the deep GEC to analyse full-batch learning with pre-learned features in Sec. 3.2. Our experiments in Sec. 4 will show perfect agreement between the theory derived using the deep GEC and simulations with deep, pre-trained generators, giving further credibility to our conjecture.

3.1. Generalisation dynamics of two-layer networks using online SGD

We first study a two-layer neural network (5) trained end-to-end using online stochastic gradient descent (6). Since the deep GEC guarantees that the local fields (λ, ν) are jointly Gaussian, permitting to express the pmse of a given student and teacher in terms of only the "order parameters" Q, R, T, v

and $\tilde{v}$ (10). In order to compute the pmse at all times during training, it is thus sufficient to track the evolution of the order parameters during training, which is the goal of this section.

We will make the crucial assumption that at each step of the algorithm, we use a previously unseen sample (x, y) to compute the updates in Eq. (6). This limit of infinite training data is variously known as online learning or one-shot/single-pass SGD. Using this assumption, the dynamics of two-layer networks in the classic teacher-student setup with i.i.d. Gaussian inputs have been analysed in seminal works by Biehl and Schwarze (1995) and Saad and Solla (1995a); see also Saad and Solla (1995b); Saad (2009) for further results and Goldt et al. (2019) for a recent proof of these equations. Here, we generalise this type of analysis to two-layer networks trained on inputs coming from the deep hidden manifold model. Note that this online-learning framework has also been used by a number of recent works studying the dynamics of networks with finite N and large hidden layer $K \to \infty$ Mei et al. (2018); Rotskoff and Vanden-Eijnden (2018); Chizat and Bach (2018); Sirignano and Spiliopoulos (2019).

We derived a closed set of integro-differential equations that describe the evolution of all order parameters using Conjecture 1. We provide a self-contained discussion of these equations here, and relegate the detailed derivation to Sec. C. Remarkably, the generator $\mathcal{G}(c)$ only enters the equations via the input-input and the input-latent covariance,

$$\Omega_{ij} = \mathbb{E}\, x_i x_j, \qquad \Phi_{ir} = \mathbb{E}\, x_i c_r. \tag{18}$$

The order parameter Q (10) can be written as $Q^{k\ell} \equiv \mathbb{E}\, \lambda^k \lambda^\ell \sim \sum w_i^k \Omega_{ij} w_j^\ell$. A key step in the analysis is to diagonalise this sum by projecting the student weights into the eigenspace of Ω (cf. Sec. C). We can then consider the integral representation

$$Q^{k\ell} = \int \mathrm{d}\mu_\Omega(\rho)\ \rho\ q^{k\ell}(\rho). \tag{19}$$

where $\mu_\Omega(\rho)$ is the spectral density of Ω (which is known and fixed at all times since it is a property of the generator $\mathcal{G}$), and $q^{kl}(\rho)$ is a density whose time evolution can be characterised in the thermodynamic limit. In the canonical teacher-student model with i.i.d. inputs x, introducing such a density is not necessary since the input-input covariance is trivial, $\Omega_{ij} = \delta_{ij}$. As we go to the thermodynamic limit $N \to \infty$, we can identify a continuous time-like parameter $t \equiv \mu/N$ and find that the density $q^{k\ell}(\rho)$ evolves according to

$$\begin{aligned}\frac{\partial q^{k\ell}(\rho)}{\partial t} = -\eta \Bigg(& \rho \sum_{j\neq k}^{K} \left[v^k v^j q^{k\ell}(\rho) h^{kj}_{(1)}(Q) + v^k v^j q^{j\ell}(\rho) h^{kj}_{(2)}(Q) \right] + \rho v^k v^k q^{k\ell}(\rho) h^{k}_{(3)}(Q) \\ & - v^k \sum_{n}^{M} \left[\rho \tilde{v}^n q^{k\ell}(\rho) h^{kn}_{(4)}(Q,R,T) + \frac{1}{\sqrt{\delta}} \tilde{v}^n r^{\ell n}(\rho) h^{kn}_{(5)}(Q,R,T) \right] \\ & + \text{all of the above with } \ell \to k, k \to \ell \Bigg) + \eta^2 \gamma v^k v^\ell h^{k\ell}_{(6)}(Q,R,T,v,\tilde{v}).\end{aligned} \tag{20}$$

where $\gamma \equiv \sum_\tau \rho_\tau / N$ and $\delta \equiv D/N$. The functions $h^{kj}_{(1)}$ etc. are scalar, non-linear functions that only involve averages over the pre-activations λ and ν such as $\mathbb{E}\, g(\nu^m) g'(\lambda^k) \lambda^j$, see Eq. (C.13). After invoking the deep GEC, these averages can be expressed in terms of the order parameters (10),

and hence the equation closes. Likewise, we also consider the projection of $\omega_i^m \equiv \sum_r \Phi_{ir}\tilde{w}_r^m$ into the eigenspace of Ω and consider the integral representation

$$R^{km} = \frac{1}{\sqrt{\delta}} \int \mathrm{d}\mu_\Omega(\rho)\; r^{km}(\rho). \tag{21}$$

We find that $r^{km}(\rho)$ evolves as

$$\frac{\partial r^{km}(\rho)}{\partial t} = -\eta v^k \left(\rho \sum_{j\neq k}^{K} \left[v^j r^{km}(\rho) h_{(1)}^{kj}(Q) + v^j \rho r^{jm}(\rho) h_{(2)}^{kj}(Q) \right] + v^k \rho r^{km}(\rho) h_{(3)}^{k}(Q) \right.$$
$$\left. - \sum_{n}^{M} \left[\rho \tilde{v}^n r^{km}(\rho) h_{(4)}^{kn}(Q,R,T) + \frac{1}{\sqrt{\delta}} \tilde{v}^n h_{(5)}^{kn}(Q,R,T) \right] \right). \tag{22}$$

Finally, the equation for v can be obtained directly from the SGD update (6) and reads

$$\frac{\mathrm{d}v^k}{\mathrm{d}t} = \eta \left[\sum_{n}^{M} \tilde{v}_n h_{(7)}^{kn}(Q,R) - \sum_{j}^{K} v^j h_{(7)}^{kj}(Q) \right]. \tag{23}$$

Discussion The importance of the spectral properties of the data was recognised for learning in linear neural networks Baldi and Hornik (1989); Le Cun et al. (1991); Krogh and Hertz (1992); Saxe et al. (2014). Yoshida and Okada (2019) extended the ODE analysis for non-linear networks to inputs with a covariance matrix having $O(1)$ non-degenerate eigenvalues, while implicitly assuming that inputs have a Gaussian distribution. Goldt et al. (2020) analysed online learning in the hidden manifold for a single-layer generator of the form $x = \sigma(Ac)$; their result also involved more order parameters than our analysis. Our approach handles a more general data structure, in the sense that inputs can have arbitrary covariance matrices Ω and Φ. More importantly, the GET (Thm. 2) rigorously guarantees that we can analyse the SGD dynamics even for inputs that are drawn from pre-trained generative models such as Eq. (12) and hence do not follow a Gaussian distribution. Our experiments in the next section show how this analysis also holds for deep, pre-trained generative models such as normalising flows (see Sec. 4 for the discussion and Fig. 4 for an example of the images generated by these models).

Solving the equations of motion The equations of motion (19-23) are valid for any choice of generator network and for any teacher and student activation functions $g(x)$ and $\tilde{g}(x)$ as long as the deep GEC holds. To solve the equations for a particular setup, one needs to estimate the covariance matrices Ω and Φ, and to evaluate the functions $h_{(1)}^{kj}$ etc. that are given in the appendix. By choosing $g(x) = \tilde{g}(x) = \mathrm{erf}(x/\sqrt{2})$, all these functions have exact analytical expressions Saad and Solla (1995a). We provide robust Monte Carlo estimators of the covariance matrices of any generative network in pyTorch Paszke et al. (2019) and a numerical implementation of the equations of motion at `https://github.com/sgoldt/gaussian-equiv-2layer`.

3.2. Full-batch analysis of learning a generalised linear model with pre-learned features

We now discuss a second task in which the deep GEC 1 can be used to give a sharp analysis of the asymptotic performance: full-batch learning with pre-learned or random features. In this

task, a batch of T i.i.d. samples $\mathcal{D}_T = \{(x^\mu, y^\mu)\}_{\mu=1}^T$ from q are projected using a *feature map* $\tilde{x} = \tilde{N}^{-1/2}\sigma(Fx) \in \mathbb{R}^{\tilde{N}}$. The restrictions that we place on the projection matrix F are exactly the same that we put on the weights of the one-layer generator A in our proof of the GET, see Sec. 2.

The features $\tilde{x}$ are is then fitted with the *generalised linear model* $\hat{y} = \phi_\theta(x) = g\left(\sum_{n=1}^{\tilde{N}} w_n \tilde{x}_n\right)$, where we can take $g(x) = \text{sign}(x)$ for a classification problem or $g(x) = x$ for regression for example. The weights $\hat{w} \in \mathbb{R}^{\tilde{N}}$ are learned by minimising the empirical risk (4). Note that a for a convex loss function ℓ, the regularised risk is strongly convex and admits one and only one solution. One interesting special case of this model are random features, since for random F, in the limit $\tilde{N} \to \infty$, the expected scalar product in feature space converges to a kernel Rahimi and Recht (2008):

$$\frac{1}{\tilde{N}}\mathbb{E}_{\mathrm{F}}\left[\sigma\left(\mathrm{F}x_1\right)^\top \sigma\left(\mathrm{F}x_2\right)\right] \underset{\tilde{N}\to\infty}{\to} K(x_1, x_2). \tag{24}$$

It is out of the scope of this work to describe this construction in full generality, and we refer the curious reader to Rahimi and Recht (2008, 2009) for details on how the kernel depends on the choice of Φ_{F}. The important point here is that studying kernel regression is equivalent to studying linear regression on feature space at $\tilde{N} \to \infty$. There has been a surge of interest in kernel methods recently, as it was shown that deep neural networks are equivalent to random features in the so-called lazy regime Jacot et al. (2018); Chizat et al. (2019).

Since the feature map $\Phi_{\mathrm{F}} = \tilde{N}^{-1/2}\sigma\left(\mathrm{F}\cdot\right)$ is pre-learned, for the purpose of the theoretical analysis it can be incorporated as an additional layer to the generative model for data: $\tilde{x} = \Phi_{\mathrm{F}}(x) = \left(\Phi_{\mathrm{F}} \circ \mathcal{G}\right)(c)$, where $\mathcal{G}$ can be any of the generative models discussed previously. With this observation in mind, without loss of generality we can restrict our attention to the study of generalised linear models with data coming from a deep generative model (which includes the feature map). Up to a rescaling, the generalised linear model is equivalent to $K = 1$ in model (5), and in this section we also restrict the analysis to $M = 1$ in eq. (2). Therefore, the target outputs are simply generated from the latent vector $c \sim \mathcal{N}(0, \mathrm{I}_D)$ as in Eq. (2), which are then fitted by the network $\phi_\theta(\tilde{x})$ by minimising the regularised empirical risk (4).

Let $\mathcal{D}_S = \{(x, y)_{\mu=1}^T\}$ be a data set with T i.i.d. samples from q. Define the sample complexity $\alpha = T/\tilde{N}$ and the latent-to-input aspect ratio $\delta = D/\tilde{N}$. As in the online analysis in Section 3.1, the deep GEC 1 can be used to write an asymptotic formula for the performance of the estimator $\phi_\theta(\tilde{x})$ in the limit where $D, T, \tilde{N} \to \infty$ and the ratios $\alpha, \delta = O(1)$:

$$\epsilon_g = \mathbb{E}_{(x,y)\sim q}\,\mathsf{pmse}(y, \hat{y}(x)) \underset{N\to\infty}{\to} \frac{1}{2}\mathbb{E}_{(\nu,\lambda)\sim\mathcal{N}(0,\Sigma)}(\tilde{g}(\nu) - g(\lambda))^2 \tag{25}$$

where $(\nu, \lambda) \sim \mathcal{N}(0, \Sigma)$ are jointly Gaussian variables with covariance $\Sigma = \begin{pmatrix} \rho & m^\star \\ m^\star & q^\star \end{pmatrix}$, and

$$\rho = \frac{1}{D}||\tilde{w}||_2^2, \qquad m^\star = \frac{1}{\sqrt{ND}}\hat{w}^\top \Phi \tilde{w}, \qquad q^\star = \frac{1}{N}\hat{w}^\top \Omega \hat{w}. \tag{26}$$

The covariances Φ, Ω are the moments of the equivalent Gaussian distribution, and were defined explicitly in eq. (18). In principle, $(m^\star, q^\star)$ should be computed from the estimator $\hat{w} \in \mathbb{R}^{\tilde{N}}$. Surprisingly, we can also use the deep GEC to derive a set of self-consistent equations with solution

giving directly $(m^\star, q^\star)$:

$$\begin{cases} \hat{V} = \alpha \mathbb{E}_{\xi\sim\mathcal{N}(0,1)} \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y \left(\frac{1-\partial_\omega \eta}{V} \right) \right] \\ \hat{q} = \alpha \mathbb{E}_{\xi\sim\mathcal{N}(0,1)} \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y \left(\frac{\eta-\omega}{V} \right)^2 \right] \\ \hat{m} = \frac{\alpha}{\sqrt{\delta}} \mathbb{E}_{\xi\sim\mathcal{N}(0,1)} \left[\int_{\mathbb{R}} \mathrm{d}y \; \partial_\omega \tilde{\mathcal{Z}}_y \left(\frac{\eta-\omega}{V} \right) \right] \end{cases} \quad \begin{cases} V = \frac{1}{\tilde{N}} \operatorname{tr} \left(\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-1} \Omega \\ q = \frac{1}{\tilde{N}} \operatorname{tr} \left[\left(\hat{q}\Omega + \hat{m}^2 \Phi\Phi^\top \right) \Omega \left(\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-2} \right] \\ m = \frac{\hat{m}}{\tilde{N}\sqrt{\delta}} \operatorname{tr} \Phi\Phi^\top \left(\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-1} \end{cases} \tag{27}$$

with:

$$\tilde{\mathcal{Z}}_y(y, \tilde{\omega}, \tilde{V}) = \int_{\mathbb{R}} \frac{\mathrm{d}x}{\sqrt{2\pi\tilde{V}}} e^{-\frac{(x-\tilde{w})^2}{2\tilde{V}}} \delta\left(y - \tilde{g}(x)\right), \quad \eta(y, \omega, V) = \underset{x\in\mathbb{R}}{\operatorname{argmin}} \left[\frac{(x-\omega)^2}{2V} + \ell(y, x) \right]$$

and $\omega = \sqrt{q}\xi, V = \rho - q, \tilde{w} = m/\sqrt{q}\xi, \tilde{V} = \rho - m^2/q$. Although this formula appears cumbersome, it only depends on scalar parameters and on the spectral distribution of $\Phi\Phi^\top$ and Ω. It therefore reduces the high-dimensional computation of ϵ_g to solving a low-dimensional system of equation which for a given generator $\mathcal{G}$, loss function ℓ and non-linearities $(g, \tilde{g})$ can be easily done by iteration. For random generators, the spectral distributions of $\Phi\Phi^\top$ and Ω can be computed analytically. But this formula also holds for the case of real, trained deep generative models, in which case the spectrum of $\Phi\Phi^\top$ and Ω are computed numerically via robust Monte-Carlo simulations exactly as in Section 3.1. Note that this result generalises the formula from Gerace et al. (2020) for a single-layer generator which was rigorously proved recently by Dhifallah and Lu (2020). Although it is an open problem to prove it rigorously in the current setting, we verified that it perfectly matches simulations for different loss functions and for all generative architectures discussed here. See Fig. 3 in Section 4 for one example. This provides another strong evidence for conjecture 1 - as it shows that a formula only depending on second order statistics is able to completely capture the asymptotic performance of random features trained on data from a trained generative model.

4. Experiments

The derivations of both the dynamical equations (19-23) for online SGD and the iterative equations (27) for full-batch learning with features rely on the deep GEC. While Theorem 2 gives verifiable conditions under which the conjecture is true for one-layer generators, it remains an open problem to establish the deep GEC rigorously. We thus conducted a set of experiments to compare the predictions for the pmse made by the theoretical results of Secs 3.1 and 3.2 to the test error measured in simulations. For the dynamical equations, this means comparing the evolution of the pmse and the order parameters obtained by (i) integrating Eqns. (19-23) and (ii) by evaluating Eq. (10) explicitly during a *single* run of SGD for a two-layer student with $K = 2$ hidden units. For the full-batch analysis, we compare the pmse obtained from iterating Eq. (27) with the result obtained by numerically minimising the empirical risk in Eq.(4) with gradient descent for a given sample complexity $\alpha = T/\tilde{N}$. For the dynamical equations, the teacher is taken to be a two-layer network with $M = 2$ hidden units, and for the full-batch learning it is taken to be a $M = 1$ generalised linear model. In both cases, the teacher weights are drawn i.i.d. from the standard normal distribution.

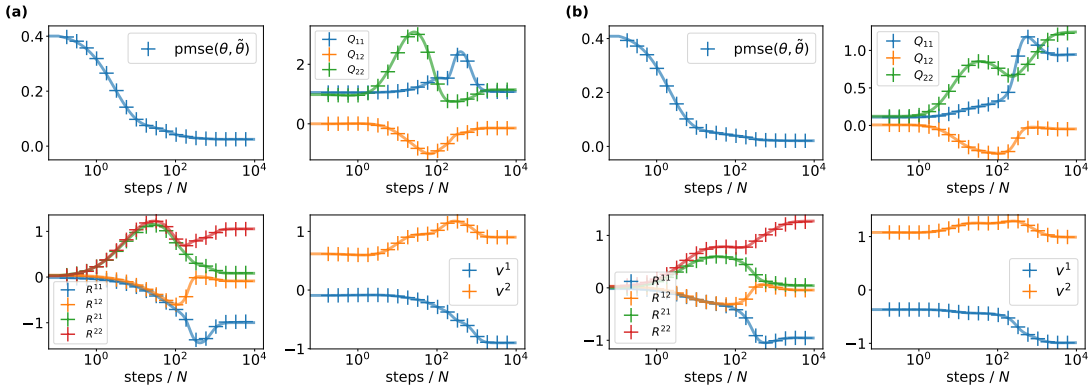

Figure 2: **Dynamics of two-layer networks: Theory vs experiments for random generators.** We compare the evolution of the pmse and the order parameters obtained from integration of Eqns. (19-23, solid lines) and a single run of SGD (crosses). **(a)** Inputs are generated by a single-layer generator (12) with i.i.d. weight matrix A and sign activation function ($D = 800, N = 8000$). **(b)** Inputs were generated by the five-layer DCGAN of Radford et al. (2016) with random weights ($D = 100, N = 3072$). *In both plots:* $M = K = 2, \tilde{v}^m = 1, \eta = 0.2, g(x) = \tilde{g}(x) = \mathrm{erf}(x/\sqrt{2})$, integration time step $\mathrm{d}t = 0.01$.

4.1. Fully-connected and convolutional generators with random weights

As a first test, we verified that the equations correctly predict the dynamics of online SGD in a setting where Theorem 2 applies: a one-layer generator $\mathcal{G}(c)$ (12) with i.i.d. weight matrix A and sign activation function. In a second set of experiments, we drew the inputs from the deep convolutional GAN (dcGAN) of Radford et al. (2016) with random i.i.d. weights. The dcGAN consists of five convolutional layers, each followed by a Batch Normalisation layer and a ReLU activation function. The final activation function is $\tanh(x)$ (see Sec. E for a detailed description). We show an example of the comparison for both generators in Fig. 2, with more runs in Sec. E. The agreement between equations and simulations in both experiments is very good.

4.2. Pre-trained deep convolutional GAN

We also used an instance of a dcGAN that was pre-trained on CIFAR100 dataset Krizhevsky et al. (2009) in grayscale, with weights provided by Singh. On the left of Fig. 3, we show 32 samples of the original dataset (top four rows) and 32 images generated by this network (bottom four rows). On the level of the replica analysis (27), the change of generator weights is reflected in the change of the covariance matrices Ω_{ij} and Φ_{ir} (18), which need to be estimated precisely. In Fig. 3 we compare the pmse at different sample complexities predicted by eq. (25) for logistic regression with Gaussian features F of different sizes with the result obtained by running gradient descent on the empirical risk. Although we didn't include the plots for conciseness, we observe the same good agreement for other tasks and for all the generative models discussed in this section.

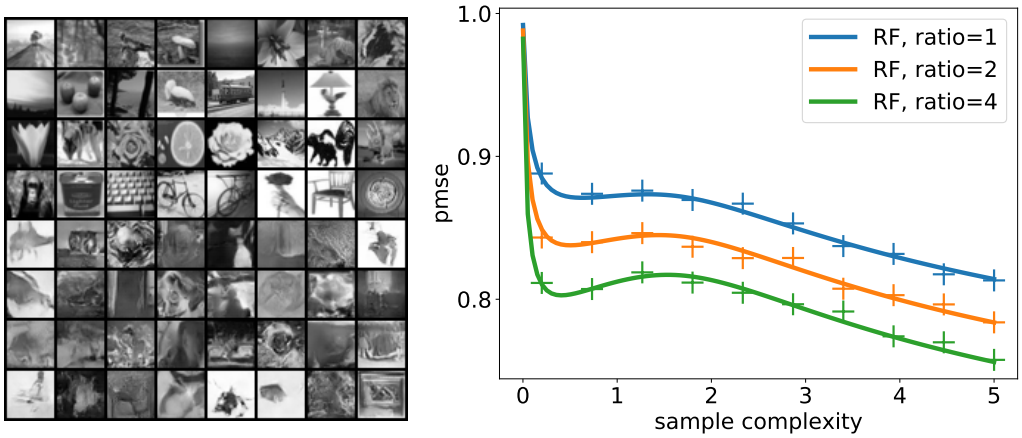

Figure 3: **Theory vs experiment for random-features logistic regression:** (*Left*) Images drawn from the CIFAR100 dataset in grayscale (top four rows) and drawn from the deep convolutional GAN trained on CIFAR100 (bottom four rows). (*Right*) Generalisation performance of Random Features logistic regression. The random features matrix $\mathrm{F} \in \mathbb{R}^{\tilde{N} \times N}$ was taken to be Gaussian and the non-linearity $\sigma = \mathrm{sign}$. The input data was generated from a dcGAN pre-trained on CIFAR100 grayscale data set Krizhevsky et al. (2009) as a function of the sample complexity $\alpha = P/\tilde{N}$ and fixed weight decay $\lambda = 10^{-2}$. Different curves correspond to different projection aspect ratios $\tilde{N}/N$.

4.3. Normalising flows: the real NVP

We finally tested the validity of the deep GEC with a generative model from the class of normalising flows Tabak et al. (2010); Tabak and Turner (2013); Rezende and Mohamed (2015); Kobyzev et al. (2020); Papamakarios et al. (2019). These models obtain a given target distribution from a series of bijective transformations of a much simpler distribution, say the multidimensional normal distribution. Constructing a probability density in this way has the advantage that the model's output distribution can be written down exactly, making it possible to minimise the exact log-likelihood. This should be contrasted with variational auto-encoders Kingma and Welling (2014), where a bound on the log-likelihood is optimised, or GANs, where the unsupervised problem of density estimation is transformed into a supervised learning problem Goodfellow et al. (2014). For the purpose of verifying the GET via the validity of the dynamical equations, normalising flows have the desirable property that their latent dimension D is equal to the dimension of the output, i.e. for CIFAR10 images, $D = N = 3072$, which is close to the regime $D, N \to \infty$ of our analysis. We trained an instance of the real NVP model of Dinh et al. (2017) using the pyTorch port of the original TensorFlow implementation provided by Mu. Using the original hyper-parameters Dinh et al. (2017), we reached an average value of ≈ 3.5 bits/dim on the validation set, which agrees with the value of 3.49 bits / dim reported there. Images generated by the trained model are shown in the bottom four rows of the grid at the bottom of Fig. 4. The comparison between ODEs and simulation (bottom right of Fig. 4) shows very good agreement between the simulation and the prediction from the ODEs,

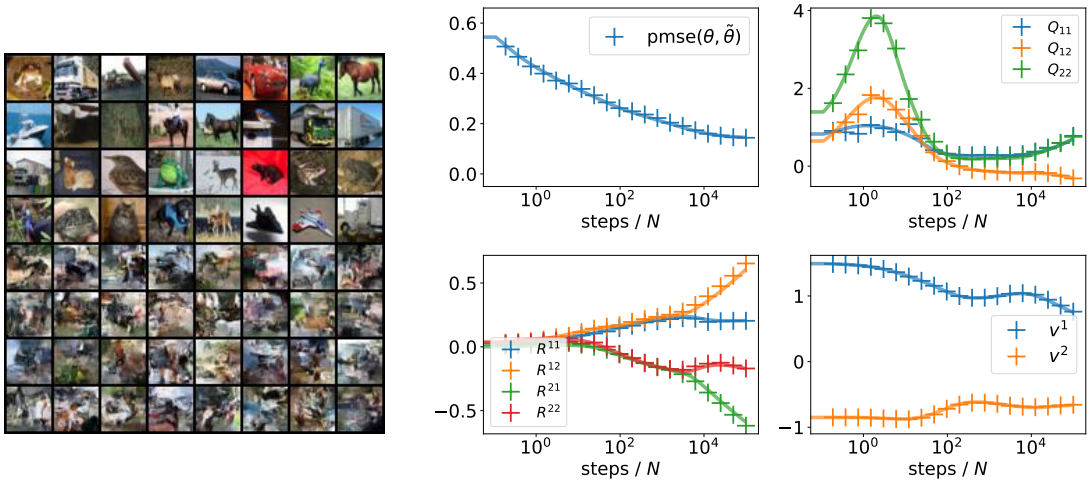

Figure 4: **Theory vs experiments for online SGD with deep, pre-trained realNVP model of Dinh et al. (2017).** *(Left)* The top four rows show images drawn randomly from the CIFAR10 data set, the bottom four rows show images drawn randomly from the realNVP model trained on CIFAR10. *(Right)* Same plot as Fig. 2 when inputs are drawn from the pre-trained realNVP. $D = N = 3072$. *In all experiments:* $M = K = 2, \tilde{v}^m = 1, \eta = 0.2, g(x) = \tilde{g}(x) = \mathrm{erf}(x/\sqrt{2})$, integration time step $\mathrm{d}t = 0.01$.

demonstrating the validity of the Gaussian Equivalence Property for this instance of a pre-trained generative model with $\sim 6.3 \cdot 10^6$ trained parameters.

Acknowledgements

We thank A. Maillard and F. Gerace for valuable discussions. We acknowledge funding from the ERC under the European Union's Horizon 2020 Research and Innovation Programme Grant Agreement 714608-SMiLe, from "Chaire de recherche sur les modèles et sciences des données", Fondation CFM pour la Recherche-ENS, and from the French National Research Agency grants ANR-17-CE23-0023-01 PAIL and ANR-19-P3IA-0001 PRAIRIE.

References

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

A. Ansuini, A. Laio, J.H. Macke, and D. Zoccolan. Intrinsic dimension of data representations in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 6109–6119, 2019.

B. Aubin, A. Maillard, J. Barbier, F. Krzakala, N. Macris, and L. Zdeborová. The committee machine: Computational to statistical gaps in learning a two-layers neural network. In *Advances in Neural Information Processing Systems 31*, pages 3227–3238, 2018.

B. Aubin, B. Loureiro, A. Maillard, F. Krzakala, and L. Zdeborová. The spiked matrix model with generative priors. In *Advances in Neural Information Processing Systems 32*, pages 8366–8377. 2019.

Y. Bahri, J. Kadmon, J. Pennington, S.S. Schoenholz, J. Sohl-Dickstein, and S. Ganguli. Statistical Mechanics of Deep Learning. *Annual Review of Condensed Matter Physics*, 11(1):501–528, 2020.

M. Baity-Jesi, L. Sagun, M. Geiger, S. Spigler, G.B. Arous, C. Cammarota, Y. LeCun, M. Wyart, and G. Biroli. Comparing Dynamics: Deep Neural Networks versus Glassy Systems. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

P. Baldi and K. Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural networks*, 2(1):53–58, 1989.

F. Benaych-Georges and R.R. Nadakuditi. The singular values and vectors of low rank perturbations of large rectangular random matrices. *Journal of Multivariate Analysis*, 111:120–135, 2012.

M. Biehl and H. Schwarze. Learning by on-line gradient descent. *J. Phys. A. Math. Gen.*, 28(3): 643–656, 1995.

S. G. Bobkov. On concentration of distributions of random weighted sums. *The Annals of Probability*, 31(1):195–215, 2003.

F. Borra, M.C. Lagomarsino, P. Rotondo, and M. Gherardi. Generalization from correlated sets of patterns in the perceptron. *Journal of Physics A: Mathematical and Theoretical*, 52(38):384004, 2019.

A. Brock, J. Donahue, and K. Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

X. Cheng and A. Singer. The spectrum of random inner-product kernel matrices. *Random Matrices: Theory and Applications*, 2(04):1350010, 2013.

L. Chizat and F. Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems 31*, pages 3040–3050, 2018.

L. Chizat, E. Oyallon, and F. Bach. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, pages 2937–2947, 2019.

SY Chung, U. Cohen, H. Sompolinsky, and D.D. Lee. Learning data manifolds with a cutting plane method. *Neural computation*, 30(10):2593–2615, 2018a.

SY Chung, Daniel D. Lee, and H. Sompolinsky. Classification and Geometry of General Perceptual Manifolds. *Physical Review X*, 8(3):31003, 2018b.

U. Cohen, SY Chung, D.D. Lee, and H. Sompolinsky. Separability and geometry of object manifolds in deep neural networks. *Nature communications*, 11(1):1–13, 2020.

T.M. Cover. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. *IEEE Transactions on Electronic Computers*, EC-14(3):326–334, 1965.

A. G. de G. Matthews, J. Hron, M. Rowland, R.E. Turner, and Z. Ghahramani. Gaussian process behaviour in wide deep neural networks. In *International Conference on Learning Representations*, 2018.

O. Dhifallah and Y. M. Lu. A precise performance analysis of learning with random features. *arXiv:2008.11904*, 2020.

P. Diaconis and D. Freedman. Asymptotics of graphical projection pursuit. *The Annals of Statistics*, 12(3):793–815, 1984.

L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using real NVP. 2017.

S. Du, J. Lee, Y. Tian, A. Singh, and B. Poczos. Gradient descent learns one-hidden-layer CNN: Don't be afraid of spurious local minima. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 1339–1348, 2018.

A. Engel and C. Van den Broeck. *Statistical Mechanics of Learning*. Cambridge University Press, 2001.

Z. Fan and A. Montanari. The spectral norm of random inner-product kernel matrices. *Probability Theory and Related Fields*, 173(1-2):27–85, 2019.

K. Fukushima and S. Miyake. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern recognition*, 15(6):455–469, 1982.

M. Gabrié. Mean-field inference methods for neural networks. *Journal of Physics A: Mathematical and Theoretical*, 53(22):223002, 2020.

M. Gabrié, A. Manoel, C. Luneau, J. Barbier, N. Macris, F. Krzakala, and L. Zdeborová. Entropy and mutual information in models of deep neural networks. In *Advances in Neural Information Processing Systems 31*, pages 1826–1836, 2018.

E. Gardner and B. Derrida. Three unfinished works on the optimal storage capacity of networks. *Journal of Physics A: Mathematical and General*, 22(12):1983–1994, 1989.

F. Gerace, B. Loureiro, F. Krzakala, M. Mézard, and L. Zdeborová. Generalisation error in learning with random features and the hidden manifold model. In *37th International Conference on Machine Learning (ICML)*, 2020.

B. Ghorbani, S. Mei, T. Misiakiewicz, and A. Montanari. Limitations of lazy training of two-layers neural network. In *Advances in Neural Information Processing Systems 32*, pages 9111–9121. 2019.

S. Goldt, M.S. Advani, A.M. Saxe, F. Krzakala, and L. Zdeborová. Dynamics of stochastic gradient descent for two-layer neural networks in the teacher-student setup. In *Advances in Neural Information Processing Systems 32*, 2019.

S. Goldt, M. Mézard, F. Krzakala, and L. Zdeborová. Modeling the influence of data structure on learning in neural networks: The hidden manifold model. *Phys. Rev. X*, 10(4):041044, 2020.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

W. Hachem, P. Loubaton, and J. Najim. Deterministic equivalents for certain functionals of large random matrices. *Ann. Appl. Probab.*, 17(3):875–930, 2007.

P. Hall and K.-C. Li. On almost linearity of low dimensional projections from high dimensional data. *The Annals of Statistics*, 21(2):867–889, 1993.

P. Hand, O. Leong, and V. Voroninski. Phase retrieval under a generative prior. In *Advances in Neural Information Processing Systems*, pages 9136–9146, 2018.

R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge university press, 2012.

H. Hu and Y.M. Lu. Universality laws for high-dimensional learning with random features. *arXiv:2009.07669*, 2020.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 448–456, 2015.

A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems 31*, pages 8571–8580, 2018.

D.P. Kingma and M. Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

I. Kobyzev, S. Prince, and M. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. doi: 10.1109/TPAMI.2020.2992934.

F. Koehler and A. Risteski. The comparative power of reLU networks and polynomial kernels in the presence of sparse latent structure. In *International Conference on Learning Representations (ICLR)*, 2019.

S. Kolouri, K. Nadjahi, U. Simsekli, R. Badeau, and G. Rohde. Generalized sliced wasserstein distances. In *Advances in Neural Information Processing Systems 32*, pages 261–272. 2019.

A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

A. Krogh and J.A. Hertz. Generalization in a linear perceptron in the presence of noise. *Journal of Physics A: Mathematical and General*, 25(5):1135, 1992.

Y. Le Cun, I. Kanter, and S.A. Solla. Eigenvalues of covariance matrices: Application to neural-network learning. *Physical Review Letters*, 66(18):2396, 1991.

Y. LeCun, B.E. Boser, J.S. Denker, D. Henderson, R.E. Howard, W.E. Hubbard, and L.D. Jackel. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404, 1990.

J. Lee, J. Sohl-Dickstein, J. Pennington, R. Novak, S. Schoenholz, and Y. Bahri. Deep neural networks as gaussian processes. In *International Conference on Learning Representations*, 2018.

C. Louart, Z. Liao, and Romain Couillet. A random matrix approach to neural networks. *The Annals of Applied Probability*, 28(2):1190–1248, 2018.

E. Meckes. Approximation of projections of random vectors. *Journal of Theoretical Probability*, 25 (2):333–352, 2010.

S. Mei and A. Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. arXiv:1908.05355, 2019.

S. Mei, A. Montanari, and P. Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

M. Mézard. Mean-field message-passing equations in the hopfield model and its generalizations. *Physical Review E*, 95(2):022117, 2017.

M. Mirza and S. Osindero. Conditional generative adversarial nets. arXiv:1411.1784, 2014.

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. MIT Press, 2012.

A. Montanari, F. Ruan, Y. Sohn, and J. Yan. The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime. arXiv:1911.01544, 2019.

E. Mossel. Deep learning and hierarchical generative models. arXiv:1612.09057, 2016.

Fangzhou Mu. Port of the original TensorFlow implementation of realNVP to pyTorch. https://github.com/fmu2/realNVP.

R.M. Neal. *Bayesian Learning for Neural Networks*. PhD thesis, University of Toronto, 1995.

R. O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

G. Papamakarios, E. Nalisnick, D.J. Rezende, S. Mohamed, and B. Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. 2019.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. 2019.

A.B. Patel, M.T. Nguyen, and R. Baraniuk. A probabilistic framework for deep learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2558–2566. Curran Associates, Inc., 2016.

J. Pennington and P. Worah. Nonlinear random matrix theory for deep learning. In *Advances in Neural Information Processing Systems*, pages 2637–2646, 2017.

A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184, 2008.

A. Rahimi and B. Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in neural information processing systems*, pages 1313–1320, 2009.

G. Reeves. Conditional central limit theorems for Gaussian projections. In *IEEE International Symposium on Information Theory*, pages 3055–3059, June 2017.

D. Rezende and S. Mohamed. Variational inference with normalizing flows. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1530–1538, Lille, France, 2015.

P. Rotondo, M. C. Lagomarsino, and M. Gherardi. Counting the learnable functions of geometrically structured data. *Phys. Rev. Research*, 2:023169, 2020.

G.M. Rotskoff and E. Vanden-Eijnden. Parameters as interacting particles: long time convergence and asymptotic error scaling of neural networks. In *Advances in Neural Information Processing Systems 31*, pages 7146–7155, 2018.

D. Saad. *On-line learning in neural networks*, volume 17. Cambridge University Press, 2009.

D. Saad and S.A. Solla. Exact Solution for On-Line Learning in Multilayer Neural Networks. *Phys. Rev. Lett.*, 74(21):4337–4340, 1995a.

D. Saad and S.A. Solla. On-line learning in soft committee machines. *Phys. Rev. E*, 52(4):4225–4243, 1995b.

A.M. Saxe, J.L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *ICLR*, 2014.

A.M. Saxe, Y. Bansal, J. Dapello, M.S. Advani, A. Kolchinsky, B.D. Tracey, and D.D. Cox. On the information bottleneck theory of deep learning. In *ICLR*, 2018.

M.E.A. Seddik, M. Tamaazousti, and R. Couillet. Kernel random matrices of large concentrated data: the example of gan-generated images. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7480–7484. IEEE, 2019.

M.E.A. Seddik, C. Louart, M. Tamaazousti, and R. Couillet. Random matrix theory proves that deep learning representations of gan-data behave as gaussian mixtures. In *37th International Conference on Machine Learning (ICML)*, 2020.

H. S. Seung, H. Sompolinsky, and N. Tishby. Statistical mechanics of learning from examples. *Physical Review A*, 45(8):6056–6091, 1992.

C. Singh. Pre-trained dcGAN model. https://github.com/csinva/gan-vae-pretrained-pytorch.

J. Sirignano and K. Spiliopoulos. Mean field analysis of neural networks: A central limit theorem. *Stochastic Processes and their Applications*, 2019.

M. Soltanolkotabi, A. Javanmard, and J.D. Lee. Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Transactions on Information Theory*, 65(2): 742–769, 2018.

V. N. Sudakov. Typical distributions of linear functionals in finite-dimensional spaces of high dimension. *Soviet Math. Doklady*, 16(6):1578–1582, 1978.

E. G Tabak and C.V. Turner. A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2):145–164, 2013.

E. G Tabak, E. Vanden-Eijnden, et al. Density estimation by dual ascent of the log-likelihood. *Communications in Mathematical Sciences*, 8(1):217–233, 2010.

Y. Tian. An analytical formula of population gradient for two-layered relu network and its applications in convergence and critical point analysis. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, page 3404–3413, 2017.

V. Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

T.L.H. Watkin, A. Rau, and M. Biehl. The statistical mechanics of learning a rule. *Reviews of Modern Physics*, 65(2):499–556, 1993.

Y. Yoshida and M. Okada. Data-dependence of plateau phenomenon in learning with neural network — statistical mechanical analysis. In *Advances in Neural Information Processing Systems 32*, pages 1720–1728, 2019.

L. Zdeborová. Understanding deep learning is also a job for physicists. *Nature Physics*, 2020.

L. Zdeborová and F. Krzakala. Statistical physics of inference: thresholds and algorithms. *Adv. Phys.*, 65(5):453–552, 2016.

K. Zhong, Z. Song, P. Jain, P.L. Bartlett, and I.S. Dhillon. Recovery guarantees for one-hidden-layer neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 4140–4149. JMLR. org, 2017.

Appendix A. Proof of the Gaussian Equivalence Theorem

There are two main steps to the proof. First we provide a one-dimensional GET (Theorem 3), which is stated under a more general setting and then we show how Theorem 2 of the main text follows as a special case.

A.1. One-dimensional GET

Let $Z = (Z_1, \dots, Z_d)$ be a vector of standard Gaussian variables and let $X = (X_1, \dots, X_n)$ be generated according to $X_i = \sigma_i(a_i^\top Z)$, $i = 1, \dots, n$, where each $\sigma_i \colon \mathbb{R} \to \mathbb{R}$ and each a_i is a unit vector in $\mathbb{R}^d$. Let ρ be the $n \times n$ positive semi-definite matrix $\rho_{ij} = a_i^\top a_j$ and let $\tilde{\rho} = \rho - I_n$ be the matrix obtained by setting the diagonal entries to zero.

The main result of this section provides a Gaussian approximation for a one-dimensional projection of X. We define I to be the subset of $[n] = \{1, \dots, n\}$ such that σ_i is not affine. Notice that the variables indexed by the complement of the set, namely $\{X_i,\ i \in [n] \backslash I\}$, are jointly Gaussian by construction.

Assumption 1 (Weak Correlation) *There exists a constant C_ρ such that*

$$\sum_{i,j \in I} \tilde{\rho}_{ij}^4 \leq C_\rho^4. \tag{A.1}$$

Assumption 2 (Smoothness) *Each σ_i is twice differentiable. Furthermore, there exists a constant C_σ such that for all $i \in I$,*

$$\max\left\{\mathbb{E}[(\sigma_i(u))^4]^{1/4}, \left(\mathbb{E}\left[(\sigma_i'(u))^2\right]\right)^{1/2}, \mathbb{E}\left[(\sigma_i''(u))^2\right]^{1/2}\right\} \leq C_\sigma, \tag{A.2}$$

where $u \sim \mathcal{N}(0,1)$.

Each σ_i can be expressed via its Hermite expansion

$$\sigma_i(u) = \sum_{k=0}^{\infty} \hat{\sigma}_i(k) h_k(u), \tag{A.3}$$

where $\hat{\sigma}_i(k)$ is the kth Hermite coefficient of σ_i and h_k is the kth (normalised) probabilist's Hermite polynomial. Note that if σ_i is affine then $\hat{\sigma}_i(k) = 0$ for $k \geq 2$.

Theorem 3 *Let P be the distribution of $\frac{1}{\sqrt{n}} \sum_{i=1}^n X_i$ and let $\hat{P}$ be the Gaussian distribution with the same mean and variance. Under Assumptions 1 and 2,*

$$d(P, \hat{P}) \leq \frac{C C_\sigma}{\sqrt{n}} \left(\delta_1 + \sqrt{n\,\delta_2} + C_\sigma(C_\rho^2 + C_\rho^3) + C_\sigma^2(1 + C_\rho^4)\right), \tag{A.4}$$

where C is a universal constant,

$$\delta_1 = \frac{1}{n} \sum_{i,j,\ell \in I} \tilde{\rho}_{ij}\tilde{\rho}_{i\ell} \left(\hat{\sigma}_j(1)\hat{\sigma}_\ell(1) + 2\rho_{j\ell}\hat{\sigma}_j(2)\hat{\sigma}_\ell(2)\right) + \frac{1}{n} \sum_{i \in I} \left(\sum_{j \in [n] \backslash I} \tilde{\rho}_{ij}\hat{\sigma}_j(1)\right)^2 \tag{A.5a}$$

$$\delta_2 = \frac{1}{n} \sum_{i,j,\ell \in I} \tilde{\rho}_{ij}^2\tilde{\rho}_{i\ell}^2 \left(2\hat{\sigma}_j(2)\hat{\sigma}_\ell(2) + 6\rho_{j\ell}\hat{\sigma}_j(3)\hat{\sigma}_\ell(3)\right) \tag{A.5b}$$

and I is the subset of $\{1, \dots, n\}$ such that σ_i is not affine.

A.2. Proof of Theorem 2

Having established the one-dimensional GET, we are now in a position to prove Theorem 2 of the main text. Let P be the distribution on $\mathbb{R}^{K+M}$ defined by the variables

$$\lambda^k = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^k x_i, \quad k = 1, \dots, K, \qquad \nu^m = \frac{1}{\sqrt{D}} \sum_{r=1}^{D} \tilde{w}_r^m c_r, \quad m = 1, \dots, M$$

where $W = (w_i^k) \in \mathbb{R}^{K \times N}$ and $\tilde{W} = (\tilde{w}_r^m) \in \mathbb{R}^{M \times D}$ are weight matrices and $c \sim \mathcal{N}(0, I_D)$ is a vector of latent Gaussian variables. Recall that $x \in \mathbb{R}^N$ is generated according to $x_i = \sigma(a_i^\top c)$ where $\sigma : \mathbb{R} \to \mathbb{R}$ is a non-linearity and each a_i is a unit vector in $\mathbb{R}^D$.

To bound the maximum-sliced distance between P and a Gaussian approximation it is sufficient to bound the difference with respect to every one-dimensional projection. Given any unit vector $\alpha \in \mathbb{R}^{K+M}$ the variable $S \sim \alpha^\top P$ is given by

$$S = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} \sum_{k=1}^{K} \alpha^k w_i^k x_i + \frac{1}{\sqrt{D}} \sum_{r=1}^{D} \sum_{m=1}^{M} \alpha^{K+m} \tilde{w}_r^m c_r. \tag{A.6}$$

We will now express this variable using the notation in Section A.1 with problem dimensions given by $d = D$ and $n = N + D$. Define $w = (w_i) \in \mathbb{R}^N$ and $\tilde{w} = (\tilde{w}_r) \in \mathbb{R}^D$ according to

$$w_i = \sum_{k=1}^{K} \alpha^k w_i^k, \qquad \tilde{w}_r = \sum_{m=1}^{M} \alpha^{K+m} \tilde{w}_{i-N}^m. \tag{A.7}$$

Letting $Z = (Z_1, \dots, Z_d)$ be a vector of i.i.d. standard Gaussian variables, the distribution of S is equal to the distribution $\frac{1}{\sqrt{n}} \sum_{i=1}^{n} X_i$ where

$$X_i = \begin{cases} \sqrt{\dfrac{n}{N}} w_i \sigma(a_i^\top Z), & 1 \le i \le N \\ \sqrt{\dfrac{n}{D}} \tilde{w}_r e_{i-N}^\top Z, & N < i \le N + D \end{cases} \tag{A.8}$$

and e_r denotes the rth standard basis vector in $\mathbb{R}^d$. Furthermore, the assumptions of Theorem 3 are satisfied where $I = \{1, \dots, N\}$ is the set of indices for which X_i is a non-affine function of Z, the constant C_σ is bounded uniformly by the assumptions on σ and the students weights, and $C_\rho = (\sum_{i \neq j} (a_i^\top a_j)^4)^{1/4}$. Applying Theorem 3 and retaining the dominant terms with respect to C_ρ, one finds that the distance between the projection of P and the projection of the Gaussian distribution $\hat{P}$ with matched first and second moments satisfies

$$d(\alpha^\top P, \alpha^\top \hat{P}) \le \tilde{C} \left(\frac{\delta_1}{\sqrt{N}} + \sqrt{\delta_2} + \frac{\sum_{i \neq j} (a_i^\top a_j)^4}{\sqrt{N}} + \frac{1}{\sqrt{N}} \right), \tag{A.9}$$

where $\tilde{C}$ is a constant that depends on the regularity assumption of σ and the maximum magnitude of the students weights and

$$\delta_1 = \frac{1}{N}\sum_{i,j,\ell=1}^{N} w_j w_\ell \tilde{\rho}_{ij}\tilde{\rho}_{i\ell}\left(\hat{\sigma}^2(1) + 2\rho_{j\ell}\hat{\sigma}^2(2)\right) + \frac{1}{D}\sum_{i=1}^{N}\sum_{r,r'=1}^{D} a_{ir}a_{ir'}\tilde{w}_r\tilde{w}_{r'} \tag{A.10}$$

$$\delta_2 = \frac{1}{N}\sum_{i,j,\ell=1}^{N} w_j w_\ell \tilde{\rho}_{ij}^2\tilde{\rho}_{i\ell}^2\left(2\hat{\sigma}^2(2) + 6\rho_{j\ell}\hat{\sigma}^2(3)\right). \tag{A.11}$$

Recalling the definitions of the matrices M_1 and M_2, it follows that

$$\frac{\delta_1}{\sqrt{N}} = O\left(\left\|\frac{1}{\sqrt{N}}w^\top M_1^{1/2}\right\|^2 + \frac{1}{\sqrt{N}}\left\|\frac{1}{\sqrt{D}}\tilde{w}^\top A^\top\right\|^2\right) \tag{A.12}$$

$$\sqrt{\delta_2} = O\left(\left\|\frac{1}{\sqrt{N}}w^\top M_2^{1/2}\right\|\right). \tag{A.13}$$

Finally, recalling the definition of $(w, \tilde{w})$ we see that the following bounds holds uniformly with respect to α:

$$\frac{\delta_1}{\sqrt{N}} = O\left(\left\|\frac{1}{\sqrt{N}}WM_1^{1/2}\right\|^2 + \frac{\hat{\sigma}(1)^2}{\sqrt{N}}\left\|\frac{1}{\sqrt{N}}WA\right\|^2 + \frac{1}{\sqrt{N}}\left\|\frac{1}{\sqrt{D}}\tilde{W}^\top A^\top\right\|^2\right) \tag{A.14}$$

$$\sqrt{\delta_2} = O\left(\left\|\frac{1}{\sqrt{N}}WM_2^{1/2}\right\|\right). \tag{A.15}$$

This completes the proof of Theorem 2.

A.3. Proof of Theorem 3

A.3.1. GAUSSIAN COMPARISON

The following results show that it is sufficient to bound the distance between P and a Gaussian distribution that has the same mean but possibly different variance.

Lemma 4 *For any $\mu \in \mathbb{R}$ and $v_1, v_2 \geq 0$,*

$$d(\mathcal{N}(\mu, v_1), \mathcal{N}(\mu, v_2)) = \frac{1}{2}\left|v_1 - v_2\right| \tag{A.16}$$

Proof Without loss of generality assume $v_1 \leq v_2$. Letting U_1, U_2 be independent standard Gaussian variables we have $X_1 = \mu + \sqrt{v_1}U_1 \sim \mathcal{N}(\mu, v_1)$ and $X_2 = X_1 + \sqrt{v_2 - v_1}U_2 \sim \mathcal{N}(\mu, v_2)$. For each $f \in \mathcal{F}$, a second order Taylor series expansion gives

$$f(X_2) - f(X_1) \leq \sqrt{v_2 - v_1}U_2 f'(X_1) + \frac{1}{2}(v_2 - v_1)U_2^2\|f''\|_\infty. \tag{A.17}$$

The first term has zero mean, because U_2 is independent of X_1. By assumption $\|f''\|_\infty \leq 1$ and thus $|\mathbb{E}\left[f(X_2)\right] - \mathbb{E}\left[f(X_1)\right]| \leq \frac{1}{2}|v_2 - v_1|$ for all $f \in \mathcal{F}$. To see that this upper bound is tight, note that the inequality is attained for the choice $f(x) = \frac{1}{2}(x - \mu)^2$. ■

Lemma 5 *Let P be a distribution on $\mathbb{R}$ with mean μ and variance v. For all $\tilde{v} \geq 0$,*

$$d(P, \mathcal{N}(\mu, v)) \leq 2d(P, \mathcal{N}(\mu, \tilde{v})). \tag{A.18}$$

Proof By the triangle inequality,

$$d(P, \mathcal{N}(\mu, v)) \leq d(P, \mathcal{N}(\mu, \tilde{v})) + d(\mathcal{N}(\mu, v), \mathcal{N}(\mu, \tilde{v})). \tag{A.19}$$

Noting that the function $f(x) = \frac{1}{2}(x-\mu)^2$ belongs to $\mathcal{F}$ the first term satisfies $d(P, \mathcal{N}(\mu, \tilde{v})) \geq \frac{1}{2}|v - \tilde{v}|$. By Lemma 4, the second term satisfies $d(\mathcal{N}(\mu, v), \mathcal{N}(\mu, \tilde{v})) = \frac{1}{2}|v - \tilde{v}|$. Combining these inequalities gives the stated result. ■

A.3.2. REPLACEMENT METHOD

We assume with without loss of generality that each X_i has zero mean and thus $\hat{\sigma}_i(0) = 0$. For the purposes of comparison, we define the Gaussian variables

$$U_i = a_i^\top Z, \qquad \hat{X}_i = \hat{\sigma}_i(1)U_i + \xi_i, \tag{A.20}$$

where $\xi_1, \ldots, \xi_n$ are independent Gaussian variables with mean zero and variance $\mathsf{Var}(\xi_i) = \mathsf{Var}(X_i) - \hat{\sigma}_i^2(1)$ chosen such that X_i and $\hat{X}_i$ have the same second moment. Notice that each U_i has mean zero, unit variance, and $\mathsf{Cov}(U_i, U_j) = \rho_{ij}$. Moreover, since $\frac{1}{\sqrt{n}}\sum_{i=1}^n \hat{X}_i$ is a Gaussian variable with the same mean as $\frac{1}{\sqrt{n}}\sum_{i=1}^n X_i$ it follows from Lemma 5 that $d(P, \hat{P}) \leq 2\sup_{f\in\mathcal{F}} \Delta(f)$ where

$$\Delta(f) = \mathbb{E}\left[f\left(\frac{1}{\sqrt{n}}\sum_{i=1}^n X_i\right) - f\left(\frac{1}{\sqrt{n}}\sum_{i=1}^n \hat{X}_i\right)\right]. \tag{A.21}$$

We use the replacement method to bound the term $\Delta(f)$. For $i = 1, \ldots, n$ define the hybrid random variable

$$S_i = \frac{1}{\sqrt{n}}\sum_{j=1}^{i-1} X_j + \frac{1}{\sqrt{n}}\sum_{j=i+1}^{n} \hat{X}_j, \tag{A.22}$$

which excludes the contribution of the ith term. Then, we obtain the telescoping sum:

$$\Delta(f) = \sum_{i=1}^n \Delta_i(f), \quad \text{where} \quad \Delta_i(f) = \mathbb{E}\left[f\left(S_i + \tfrac{1}{\sqrt{n}}X_i\right) - f\left(S_i + \tfrac{1}{\sqrt{n}}\hat{X}_i\right)\right]. \tag{A.23}$$

The next result provides a useful bound on $\Delta_i(f)$ in terms of auxiliary random variables.

Lemma 6 *Let (A_i, B_i) be a pair of random variables that is independent of (U_i, ξ_i). Then,*

$$\Delta_i(f) \leq \frac{CK_i}{\sqrt{n}}\left(\mathbb{E}\left[B_i^2\right] + \mathbb{E}\left[(S_i - A_i)^2\right] + \sqrt{\mathbb{E}\left[(S_i - A_i - B_iU_i)^2\right]} + \frac{K_i^2}{n}\right) \tag{A.24}$$

where C is a universal constant and $K_i = (\mathbb{E}\left[X_i^4\right])^{1/4}$.

Proof

For any real numbers s, x, y, a third order Taylor series expansion of f about s yields

$$\left|f(s+x) - f(s+y) - (x-y)f'(s) - (x^2-y^2)f''(s)\right| \leq \frac{1}{6}(|x|^3 + |y|^3)\|f'''\|_\infty. \tag{A.25}$$

Furthermore, for any real numbers a, b, u, we can write

$$\left|f'(s) - f'(a+bu)\right| \leq |s-a-bu|\|f''\|_\infty \tag{A.26}$$

$$\left|f'(s) - f'(a) - buf''(a)\right| \leq |s-a-bu|\|f''\|_\infty + \frac{1}{2}(bu)^2\|f'''\|_\infty \tag{A.27}$$

$$\left|f''(s) - f''(a)\right| \leq |s-a|\|f'''\|_\infty. \tag{A.28}$$

Combining the above displays with the assumption $\|f''\|_\infty, \|f'''\|_\infty \leq 1$ yields

$$\begin{aligned} f(s+x) - f(s+y) \leq (x-y)\left[f'(a) + buf''(a)\right] + (x^2-y^2)f''(a) + |x-y|\,|s-a-bu| \\ + \frac{1}{2}|x-y|\,(bu)^2 + |x^2-y^2|\,|s-a| + \frac{1}{6}\left(|x|^3 + |y)|^3\right). \end{aligned} \tag{A.29}$$

Evaluating this inequality with (a, b, s, u, x, y) replaced by $(A_i, B_i, S_i, U_i, \frac{1}{\sqrt{n}}X_i, \frac{1}{\sqrt{n}}\hat{X}_i)$ and then taking the expectation of both sides leads to

$$\begin{aligned} \Delta_i(f) \leq \frac{1}{\sqrt{n}}\mathbb{E}\left[X_i - \hat{X}_i\right]\mathbb{E}\left[f'(A_i)\right] + \frac{1}{\sqrt{n}}\mathbb{E}\left[(X_i - \hat{X}_i)U_i\right]\mathbb{E}\left[B_i f''(A_i)\right] \\ + \frac{1}{n}\mathbb{E}\left[(X_i^2 - \hat{X}_i^2)\right]\mathbb{E}\left[f''(A_i)\right] + \frac{1}{\sqrt{n}}\mathbb{E}\left[|X_i - \hat{X}_i|\,|S_iA_i - B_iU_i|\right] \\ + \frac{1}{2\sqrt{n}}\mathbb{E}\left[|X_i - \hat{X}_i|U_i^2\right]\mathbb{E}\left[B_i^2\right] + \frac{1}{n}\mathbb{E}\left[|X_i^2 - \hat{X}_i^2|\,|S_i - A_i|\right] \\ + \frac{1}{6n^{3/2}}\left(\mathbb{E}\left[|X_i|^3\right] + \mathbb{E}\left[|\hat{X}_i|^3\right]\right). \end{aligned} \tag{A.30}$$

Here, we have used the independence between (A_i, B_i) and (U_i, ξ_i) to factorise the expectations. By the construction of $\hat{X}_i$ the first three terms on the right-hand side are zero. Using the Cauchy-Schwarz inequality and the Jensen's inequality, the upper bound can be simplified as follows:

$$\begin{aligned} \Delta_i(f) \leq \frac{1}{\sqrt{n}}\sqrt{\mathbb{E}\left[(X_i - \hat{X}_i)^2\right]\mathbb{E}\left[(S_iA_i - B_iU_i)^2\right]} + \frac{1}{2\sqrt{n}}\sqrt{\mathbb{E}\left[(X_i - \hat{X}_i)^2\right]\mathbb{E}\left[U_i^4\right]\mathbb{E}\left[B_i^2\right]} \\ + \frac{1}{n}\sqrt{\mathbb{E}\left[(X_i^2 - \hat{X}_i^2)^2\right]\mathbb{E}\left[(S_i - A_i)^2\right]} + \frac{1}{6n^{3/2}}\left(\mathbb{E}\left[|X_i|^3\right] + \mathbb{E}\left[|\hat{X}_i|^3\right]\right). \end{aligned} \tag{A.31}$$

From the construction of $\hat{X}_i$ it is straightforward to verify that

$$\mathbb{E}\left[(X_i - \hat{X}_i)^2\right] \leq C_1K_i^2, \quad \mathbb{E}\left[(X_i^2 - \hat{X}_i^2)^2\right] \leq C_2K_i^4, \quad \left(\mathbb{E}\left[|X_i|^3\right] + \mathbb{E}\left[|\hat{X}_i|^3\right]\right) \leq C_3K_i^3$$

for universal constants C_1, C_2, C_3, and thus

$$\Delta_i(f) \leq \frac{CK_i}{\sqrt{n}}\left(\mathbb{E}\left[B_i^2\right] + \frac{K_i}{\sqrt{n}}\sqrt{\mathbb{E}\left[(S_i - A_i)^2\right]} + \sqrt{\mathbb{E}\left[(S_i - A_i - B_iU_i)^2\right]} + \frac{K_i^2}{n}\right). \tag{A.32}$$

Finally, by the basic inequality $xy \leq \frac{1}{2}(x^2 + y^2)$ we have

$$\frac{K_i}{\sqrt{n}}\sqrt{\mathbb{E}\left[(S_i - A_i)^2\right]} \leq \frac{K_i^2}{2n} + \frac{1}{2}\mathbb{E}\left[(S_i - A_i)^2\right], \tag{A.33}$$

and combining the last two displays gives the stated bound. ■

A.3.3. DECOMPOSITION ARGUMENT

In view of Lemma 6, the next question is how to specify the variables (A_i, B_i). We use a decomposition argument that leverages the Gaussianity of U. Let i be fixed and for each $j \neq i$ define the Gaussian variables $\tilde{U}_j = U_j - \rho_{ij}U_i$. Note that U_i and $\tilde{U}_j$ are uncorrelated and thus independent. Further define $V_i = (\tilde{U}_1, \dots, \tilde{U}_{i-1}, \tilde{U}_{i+1}, \dots, \tilde{U}_n, \xi_1, \dots, \xi_{i-1}, \xi_{i+1}, \dots, \xi_n)$. Then, we can write $S_i = g_i(U_i, V_i)$ where

$$g_i(U_i, V_i) = \frac{1}{\sqrt{n}}\sum_{j=1}^{i-1}\sigma_j(\rho_{ij}U_i + \tilde{U}_j) + \frac{1}{\sqrt{n}}\sum_{j=i+1}^{n}\left(\hat{\sigma}_j(1)(\rho_{ij}U_i + \tilde{U}_j) + \xi_j\right). \tag{A.34}$$

Since V_i is independent of (U_i, ξ_i) we can define (A_i, B_i) as a function of V_i. Specially, we define the variables to be the first and second Hermite coefficients of the mapping $u \mapsto g_i(u, V_i)$:

$$A_i = \mathbb{E}\left[S_i \mid V_i\right] = \hat{g}_i(0; V_i), \qquad B_i = \mathbb{E}\left[U_i S_i \mid V_i\right] = \hat{g}_i(1; V_i). \tag{A.35}$$

By Gaussian integration by parts, we can also write $B_i = \mathbb{E}\left[g_i'(U_i, V_i) \mid V_i\right]$ where $g_i'(u, v)$ denotes the partial derivative with respect to the first argument. In conjunction with Jensen's inequality, we obtain the following upper bound:

$$\mathbb{E}\left[B_i^2\right] = \mathbb{E}\left[\mathbb{E}\left[g_i'(U_i, V_i) \mid V_i\right]^2\right] \leq \mathbb{E}\left[(g_i'(U_i, V_i))^2\right]. \tag{A.36}$$

Lemma 7 *Let $U \sim \mathcal{N}(0, 1)$ and let $g\colon \mathbb{R} \to \mathbb{R}$ be a twice differentiable with $\mathbb{E}\left[g^2(U)\right] < \infty$. Then,*

$$\mathbb{E}\left[(g(U) - \hat{g}(0))^2\right] \leq \mathbb{E}\left[(g'(U))^2\right] \tag{A.37}$$

$$\mathbb{E}\left[(g(U) - \hat{g}(0) - \hat{g}(1)U)^2\right] \leq \mathbb{E}\left[(g''(U))^2\right]. \tag{A.38}$$

Proof The first inequality is the Gaussian Poincaré inequality. For the second inequality we use the Plancherel formula (O'Donnell, 2014, Proposition 11.36) to write

$$\mathbb{E}\left[(g(U) - \hat{g}(0) - \hat{g}(1)U)^2\right] = \sum_{k=2}^{\infty}\hat{g}(k)^2 \leq \frac{1}{\sqrt{2}}\sum_{k=0}^{\infty}\widehat{g''}(k)^2 = \frac{1}{\sqrt{2}}\mathbb{E}\left[(g''(Z))^2\right] \tag{A.39}$$

where the third step follows from the relation $\widehat{g''}(k) = \sqrt{k+1}\sqrt{k+2}\hat{g}(k+2)$ for non-negative inters k. ■

Using Lemma 7, we obtain

$$\mathbb{E}\left[(S_i - A_i)^2\right] \leq \mathbb{E}\left[(g'(U_i, V_i))^2\right], \quad \mathbb{E}\left[(S_i - A_i - B_iU_i)^2\right] \leq \mathbb{E}\left[(g''(U_i, V_i))^2\right]. \tag{A.40}$$

Combining Lemma 6 with and (A.36) and (A.40) yields

$$\Delta_i(f) \leq \frac{CK_i}{\sqrt{n}}\left(\mathbb{E}\left[(g_i'(U_i, V_i))^2\right] + \sqrt{\mathbb{E}\left[(g_i''(U_i, V_i))^2\right]} + \frac{K_i^2}{n}\right). \tag{A.41}$$

Lemma 8 *Under Assumptions 1 and 2,*

$$\mathbb{E}\left[(g_i'(U_i,V_i))^2\right] \le \left(\frac{1}{\sqrt{n}}\sum_{j\in[n]}\tilde{\rho}_{ij}\hat{\sigma}_j(1)\right)^2 + \frac{2}{n}\sum_{j_1,j_2\in[i]}\tilde{\rho}_{ij_1}\tilde{\rho}_{ij_2}\tilde{\rho}_{j_1j_2}\hat{\sigma}_{j_1}(2)\hat{\sigma}_{j_2}(2) + \frac{C_\sigma^2(1+C_\rho^2)}{n}\sum_{j\in I}\tilde{\rho}_{ij}^2 \tag{A.42}$$

$$\mathbb{E}\left[(g''(U_i,V_i))^2\right] \le \left(\frac{\sqrt{2}}{\sqrt{n}}\sum_{j\in[i]}\tilde{\rho}_{ij}^2\hat{\sigma}_j(2)\right)^2 + \frac{6}{n}\sum_{j_1,j_2\in[i]}\tilde{\rho}_{ij_1}^2\tilde{\rho}_{ij_2}^2\tilde{\rho}_{j_1j_2}\widehat{\sigma}_{j_1}(3)\widehat{\sigma}_{j_2}(3) + \frac{C_\sigma^2(1+C_\rho^2)}{n}\sum_{j\in I}\tilde{\rho}_{ij}^4 \tag{A.43}$$

Proof Recalling that $\tilde{\rho}_{ij} = \rho_{ij}\mathbf{1}_{i\neq j}$ and using the relation $\hat{\sigma}_j(1) = \mathbb{E}\left[\sigma_j'(U_j)\right]$ leads to

$$g_i'(U_i,V_i) = \frac{1}{\sqrt{n}}\sum_{j\in[i]}\tilde{\rho}_{ij}(\sigma_j'(U_j) - \mathbb{E}\left[\sigma_j'(U_j)\right]) + \frac{1}{\sqrt{n}}\sum_{j\in[n]}\tilde{\rho}_{ij}\hat{\sigma}_j(1). \tag{A.44}$$

Because the first term has zero mean and the second term is non-random, it follows that

$$\mathsf{Var}(g_i'(U_i,V_i)) = \frac{1}{n}\sum_{j_1,j_2\in[i]\,:\,j_1\neq j_2}\tilde{\rho}_{ij_1}\tilde{\rho}_{ij_2}\,\mathsf{Cov}(\sigma_{j1}'(U_j),\sigma_{j_2}'(U_j)) + \frac{1}{n}\sum_{j\in[i]}\tilde{\rho}_{ij}^2\mathsf{Var}(\sigma_j'(U_j)).$$

Expanding the covariance in terms of the Hermite coefficients yields

$$\mathsf{Cov}(\sigma_{j1}'(U_{j_1}),\sigma_{j_2}'(U_{j_2})) = \sum_{k=1}^{\infty}\rho_{j_1j_2}\widehat{\sigma'}_{j_1}(k)\widehat{\sigma'}_{j_2}(k) \tag{A.45}$$

$$\le \rho_{j_1j_2}\widehat{\sigma'}_{j_1}(k)\widehat{\sigma'}_{j_2}(k) + \rho_{j_1j_2}^2\sum_{k=1}^{\infty}|\widehat{\sigma'}_{j_1}(k)\widehat{\sigma'}_{j_2}(k)| \tag{A.46}$$

$$\le 2\rho_{j_1j_2}\widehat{\sigma'}_{j_1}(k)\widehat{\sigma'}_{j_2}(k) + \rho_{j_1j_2}^2\sqrt{\mathsf{Var}(\sigma_{j1}'(U_{j_1})\mathsf{Var}(\sigma_{j1}'(U_{j_2})} \tag{A.47}$$

where the last line follows from $\widehat{\sigma'}_j(1) = \sqrt{2}\widehat{\sigma}_j(2)$ and the Cauchy-Schwarz inequality. Since $\mathsf{Var}(\sigma_j'(U_j)$ is equal to zero if σ_j is affine and bounded by C_σ^2 otherwise, we can write

$$\mathsf{Var}(g_i'(U_i,V_i)) \le \frac{2}{n}\sum_{j_1,j_2\in[i]}\tilde{\rho}_{ij_1}\tilde{\rho}_{ij_2}\tilde{\rho}_{j_1j_2}\hat{\sigma}_{j_1}(2)\hat{\sigma}_{j_2}(2) + \frac{C_\sigma^2}{n}\sum_{j_1,j_2\in I}|\tilde{\rho}_{ij_1}\tilde{\rho}_{ij_2}|\tilde{\rho}_{j_1j_2}^2 + \frac{C_\sigma^2}{n}\sum_{j\in I}\tilde{\rho}_{ij}^2.$$

Finally, by the Cauchy-Schwarz inequality, the second term can be simplified as follows:

$$\sum_{j_1,j_2\in I}\tilde{\rho}_{ij_1}\tilde{\rho}_{ij_2}\tilde{\rho}_{j_1j_2}^2 \le \sqrt{\sum_{j_1,j_2\in I}\tilde{\rho}_{ij_1}^2\tilde{\rho}_{ij_2}^2}\sqrt{\sum_{j_1,j_2\in I}\tilde{\rho}_{j_1j_2}^4} \le C_\rho^2\sum_{j\in I}\tilde{\rho}_{ij}^2 \tag{A.48}$$

Using a similar approach for $g_i''(U_i, V_i)$ and noting that $\widehat{\sigma''}_j(0) = \sqrt{2}\hat{\sigma}_j(2)$ and $\widehat{\sigma''}_j(1) = \sqrt{6}\hat{\sigma}_j(3)$ leads to

$$\begin{aligned}
\mathbb{E}\left[g_i''(U_i, V_i)\right] &= \frac{\sqrt{2}}{\sqrt{n}} \sum_{j\in[i]} \tilde{\rho}_{ij}^2 \hat{\sigma}_j(2) \\
\mathsf{Var}(g_i''(U_i, V_i)) &= \frac{1}{n} \sum_{j_1, j_2 \in [i] \,:\, j_1 \neq j_2} \tilde{\rho}_{ij_1}^2 \tilde{\rho}_{ij_2}^2 \,\mathsf{Cov}(\sigma_{j1}''(U_j), \sigma_{j_2}''(U_j)) + \frac{1}{n} \sum_{j\in[i]} \tilde{\rho}_{ij}^4 \mathsf{Var}(\sigma_j''(U_j)) \\
&\leq \frac{1}{n} \sum_{j_1, j_2 \in [i]} \tilde{\rho}_{ij_1}^2 \tilde{\rho}_{ij_2}^2 \tilde{\rho}_{j_1 j_2} \widehat{\sigma''}_{j_1}(1) \widehat{\sigma''}_{j_2}(1) + \frac{C_\sigma^2}{n} \sum_{j_1, j_2 \in I} \tilde{\rho}_{ij_1}^2 \tilde{\rho}_{ij_2}^2 \tilde{\rho}_{j_1 j_2}^2 + \frac{C_\sigma^2}{n} \sum_{j\in I} \tilde{\rho}_{ij}^4 \\
&\leq \frac{6}{n} \sum_{j_1, j_2 \in [i]} \tilde{\rho}_{ij_1}^2 \tilde{\rho}_{ij_2}^2 \tilde{\rho}_{j_1 j_2} \widehat{\sigma}_{j_1}(3) \widehat{\sigma}_{j_2}(3) + \frac{C_\sigma^2 (1 + C_\rho^2)}{n} \sum_{j\in I} \tilde{\rho}_{ij}^4
\end{aligned}$$

■

A.3.4. FINAL STEPS IN PROOF

In view of (A.23), (A.41), and Lemma 8, we have all the ingredients needed to bound $\Delta(f)$. To simplify the analysis, observe that the replacement method can be applied with respect to any permutation π of the problem indices $[n]$. Averaging over all possible permutations of π of $[n]$ we can write

$$\Delta(f) = \frac{1}{n!} \sum_{\pi} \sum_{i=1}^{n} \Delta_{i,\pi}(f) \tag{A.49}$$

where $\Delta_{i,\pi}(f)$ is defined with respect to the permuted variables $(X_{\pi(1)}, \ldots, X_{\pi(n)})$. Swapping expectation over π and the summation over i, and combining with (A.41) and Lemma 8 we obtain an bound that holds uniformly for all i:

$$\frac{1}{n!} \sum_{\pi} \Delta_{i,\pi}(f) \leq \frac{1}{n} \frac{C C_\sigma}{\sqrt{n}} \left(\delta_1 + C_\sigma^2 (1 + C_\rho^2) \frac{1}{n} \sum_{i,j\in I} \tilde{\rho}_{ij}^2 + \sqrt{n\delta_2 + C_\sigma^2 (1 + C_\rho^2) C_\rho^4} + C_\sigma^2 \right). \tag{A.50}$$

Noting that $\sum_{i,j\in I} \tilde{\rho}_{ij}^2 \leq n C_\rho^2$ and simplifying the dependence on the constants C_σ, C_ρ gives the stated result. This concludes thee proof of Theorem 3

Appendix B. Conditions for the GET

In this appendix we explore the conditions for the Gaussian equivalence theorem in more detail. For an $N \times D$ matrix A, we define the symmetric $N \times N$ matrices $\rho \equiv AA^\top$ and $\tilde{\rho} \equiv AA^\top - \mathrm{I}_N$. Then, the matrices M_1 and M_2 appearing in Theorem 2 can be expressed as

$$M_1 = \hat{\sigma}^2(1) K_{11} + \hat{\sigma}^2(2) K_{21} \tag{B.1}$$

$$M_2 = \hat{\sigma}^2(2) K_{21} + \hat{\sigma}^2(3) K_{22}, \tag{B.2}$$

where

$$K_{11} = \frac{1}{\sqrt{N}}\,\tilde{\rho}^2 \tag{B.3}$$

$$K_{12} = \frac{1}{\sqrt{N}}\,\tilde{\rho}^2 \circ \rho \tag{B.4}$$

$$K_{21} = (\tilde{\rho} \circ \tilde{\rho})^2 \tag{B.5}$$

$$K_{22} = (\tilde{\rho} \circ \tilde{\rho})^2 \circ \rho \tag{B.6}$$

These matrices are positive definite by the Schur product theorem (Horn and Johnson, 2012, Sec. 7.5), and thus have positive real eigenvalues. We are interested in how the leading eigenvalues and eigenvectors depend on A.

To gain insight into the scaling behaviour of the matrices, we consider a setting where the entries of A are i.i.d. according to

$$A_{ij} = \frac{1}{\sqrt{D}}\left(\mu + \sqrt{1-\mu^2}\,Z_{ij}\right)$$

where $\mu \in [0,1]$ is a deterministic parameter and $\{Z_{ij}\}$ are i.i.d. standard Gaussian variables. The normalisation by $1/\sqrt{D}$ ensures that the column norms of A converges to one almost surely as $D \to \infty$.

B.1. Deterministic setting

In the limit where N is fixed and $D \to \infty$, it follows from the law of large numbers that $\rho = AA^\top$ converges almost surely to the deterministic $N \times N$ matrix given by

$$\rho = \mu^2 \mathbf{1}_{N\times N} + (1-\mu^2)\mathrm{I}_N. \tag{B.7}$$

Notice that this is the same matrix given Example 2 with $\mu^2 = c/\sqrt{N}$. The matrices K_{ij} can be computed exactly as

$$K_{11} = \frac{\mu^4}{\sqrt{N}}\left((N-2)\mathbf{1}_{N\times N} + \mathrm{I}_N\right) \tag{B.8a}$$

$$K_{12} = \frac{\mu^4}{\sqrt{N}}\left((N-2)\mu^2\mathbf{1}_{N\times N} + [(N-2)(1-\mu^2)+1]\mathrm{I}_N\right) \tag{B.8b}$$

$$K_{21} = \mu^4 N^{1/2} K_{11} \tag{B.8c}$$

$$K_{22} = \mu^4 N^{1/2} K_{12}. \tag{B.8d}$$

Since each of these matrices can be expressed as a weighted sum of the all ones matrix and the identity matrix, their eigenvalue decompositions can be described using using the following elementary result.

Lemma 9 *If $K = \alpha \mathbf{1}_{N\times N} + \beta \mathrm{I}_N$ for real numbers α, β with $\alpha \geq 0$, then the leading eigenvector of K is proportional to the all ones vector and the ordered real eigenvalues $\lambda_1(K) \geq \lambda_2(K) \geq \cdots \geq \lambda_N(K)$ are given by*

$$\lambda_i(K) = \begin{cases} \alpha N + \beta, & i = 1 \\ \beta, & i \geq 2 \end{cases} \tag{B.9}$$

Table 1: Leading order terms for the eigenvalues of the matrices in (B.8).

	K_{11}	K_{12}	K_{21}	K_{22}
maximum eigenvalue	$\mu^4 N^{3/2}$	$\mu^6 N^{3/2} + \mu^4 N^{1/2}$	$\mu^8 N^2$	$\mu^{10} N^2 + \mu^8 N$
2nd largest eigenvalue	$\mu^4 N^{-1/2}$	$\mu^4 N^{1/2}$	μ^8	$\mu^8 N$

By Lemma 9, each of the K_{ij} matrices has a leading eigenvector that is proportional to the all ones vector. Furthermore, the leading order terms in the eigenvalues are summarised in the Table 1 as a function of N and μ. Here, we see that if the mean parameter satisfies $\mu = O(N^{-\beta})$ for a fixed constant $\beta > 1/8$ then all of the eigenvalues except for the maximum converge to zero as $N \to \infty$. In other words, the GET holds provided that the weights are orthogonal to the all ones vector.

Evaluating with $\mu^2 = c/\sqrt{N}$ for fixed constant c (or equivalently $\beta = 1/4$) recovers the scalings given in Example 2.

B.2. Fixed aspect ratio

Next we consider the setting where $D/N \to \delta \in (0, \infty)$. Note that A can be expressed as a rank-one perturbation of an $N \times D$ matrix with i.i.d. entries. In the high dimensional setting $N \to \infty$, the asymptotic distribution of the singular values and singular vectors are given by Benaych-Georges and Nadakuditi (2012). In particular, the maximum eigenvalue satisfies

$$\lambda_1(AA^\top) \to \begin{cases} \dfrac{(1-\mu^2+\mu^2 N)((1-\mu^2)/\delta + \mu^2 N)}{\mu^2 N}, & \dfrac{\mu^2 N}{1-\mu^2} \geq \delta^{-1/2} \\ (1-\mu^2)\left(1+\sqrt{1/\delta}\right)^2, & \text{otherwise} \end{cases}, \tag{B.10}$$

and the asymptotic empirical distribution of the remaining eigenvalues converges almost surely to the Marchenko-Pastur distribution. Based on these results, the leading order terms in the first and second eigenvalues of K_{11} satisfy the following bounds almost surely:

$$\lambda_1(K_{11}) = O\left([\delta^{-1} + \delta^{-2}]N^{-1/2} + \mu^4 N^{3/2}\right) \tag{B.11}$$

$$\lambda_2(K_{11}) = O\left([\delta^{-2} + \delta^{-1} + \mu^4]N^{-1/2}\right). \tag{B.12}$$

Notice that the $\delta \to \infty$ limit of these conditions recovers the scaling given in Table 1.

The scaling behaviour of the matrices K_{12}, K_{21}, and K_{22} is more difficult to characterise theoretically because these matrices involve the Hadamard product of random matrices. In the following section we explore their behaviour numerically. For fixed δ and $\mu = O(N^{-\beta})$ we make the following observations:

- Fig. 5 shows the empirical scaling of the eigenvalues for the case $\mu = 0$ (which corresponds to Example 1) and $\mu = O(1/\sqrt{n})$. In both cases, we see that all of the eigenvalues converge to zero expect for the maximum eigenvalue of K_{21} which is order one. Moreover, the rate of convergence appears to be the same for these two cases.

- Fig. 6 shows the empirical scaling of the eigenvalues for $\beta \in \{1/5, 1/6, 1/7, 1/8\}$. For $\beta \geq 1/6$ the second largest eigenvalues of all matrices appear to be decreasing with N.

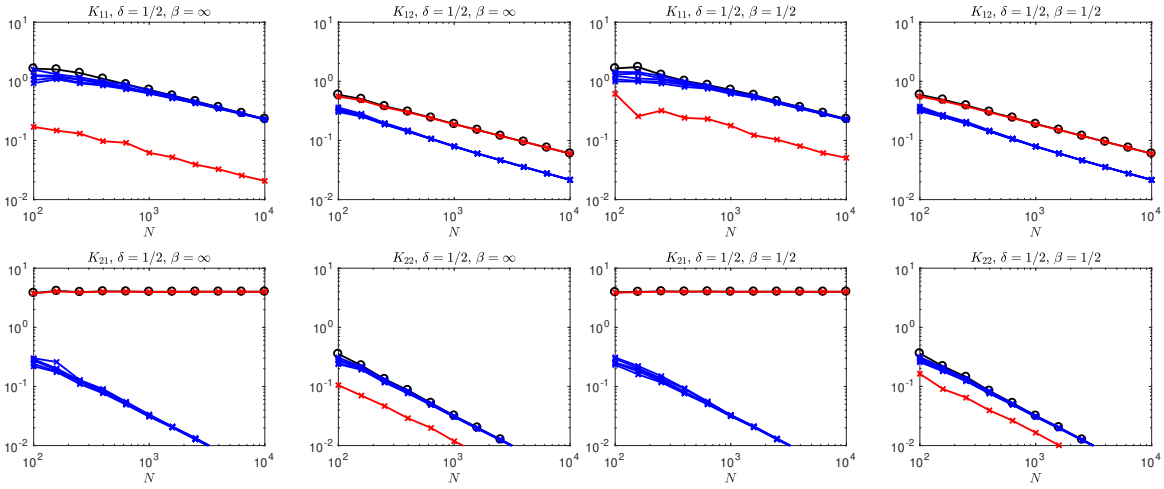

Figure 5: Scaling of eigenvalues in the K matrices. The black line is maximum eigenvalue λ_1. Blue lines are λ_i for $i \in \{2, 6\}$. The red line is the correlation with the all ones matrix; when this value is close to the maximum eigenvalue it means the leading eigenvector is close to the all ones vectors. The left panel is the case $\mu = 0$ and the right panel is the case $\mu = O(N^{-1/2})$.

However, for $\beta = 1/7$ the eigenvalues in K_{12} do not appear to be decreasing (at least for the scale of N shown) and this suggests that the conditions on μ needed to ensure convergence are more stringent then in the deterministic setting ($\delta \to \infty$) for which the condition $\beta > 1/8$ is sufficient.

Appendix C. Derivation of the equations of motion of Sec. 3.1

Here we give a detailed derivation of the equations of motion that describe the dynamics of the two-layer neural net studied in Sec. 3.1. We refer to this section for a detailed description of the setup. The GEP allows us to express the prediction mean-squared error pmse as a function of the second-layer weights v and $\tilde{v}$ as well as the second moments of (λ, ν), which we can write in terms of the covariance matrices $\Omega_{ij} = \mathbb{E}\, x_i x_j$ and $\Phi_{ir} = \mathbb{E}\, x_i c_r$ as

$$Q^{k\ell} \equiv \mathbb{E}\, \lambda^k \lambda^\ell = \frac{1}{N} \sum_{i,j}^{N} w_i^k \Omega_{ij} w_j^k, \qquad R^{km} \equiv \mathbb{E}\, \lambda^k \nu^m = \frac{1}{\sqrt{\delta}} \frac{1}{N} \sum_{i,r} w_i^k \Phi_{ir} \tilde{w}_r^m$$
$$T^{mn} \equiv \mathbb{E}\, \nu^m \nu^n = \frac{1}{D} \sum_{r,s}^{D} \tilde{w}_r^m \tilde{w}_r^n. \tag{C.1}$$

We will adopt the notational convention for tensors such as $Q^{k\ell}$ that extensive indices (taking values up to D, N) are below the line, while we'll use upper indices when they take a finite number of values up to M or K. The challenge of controlling the learning in the thermodynamic limit will be to write closed equations using matrices with only "upper" indices left. Finally, we will adopt the convention

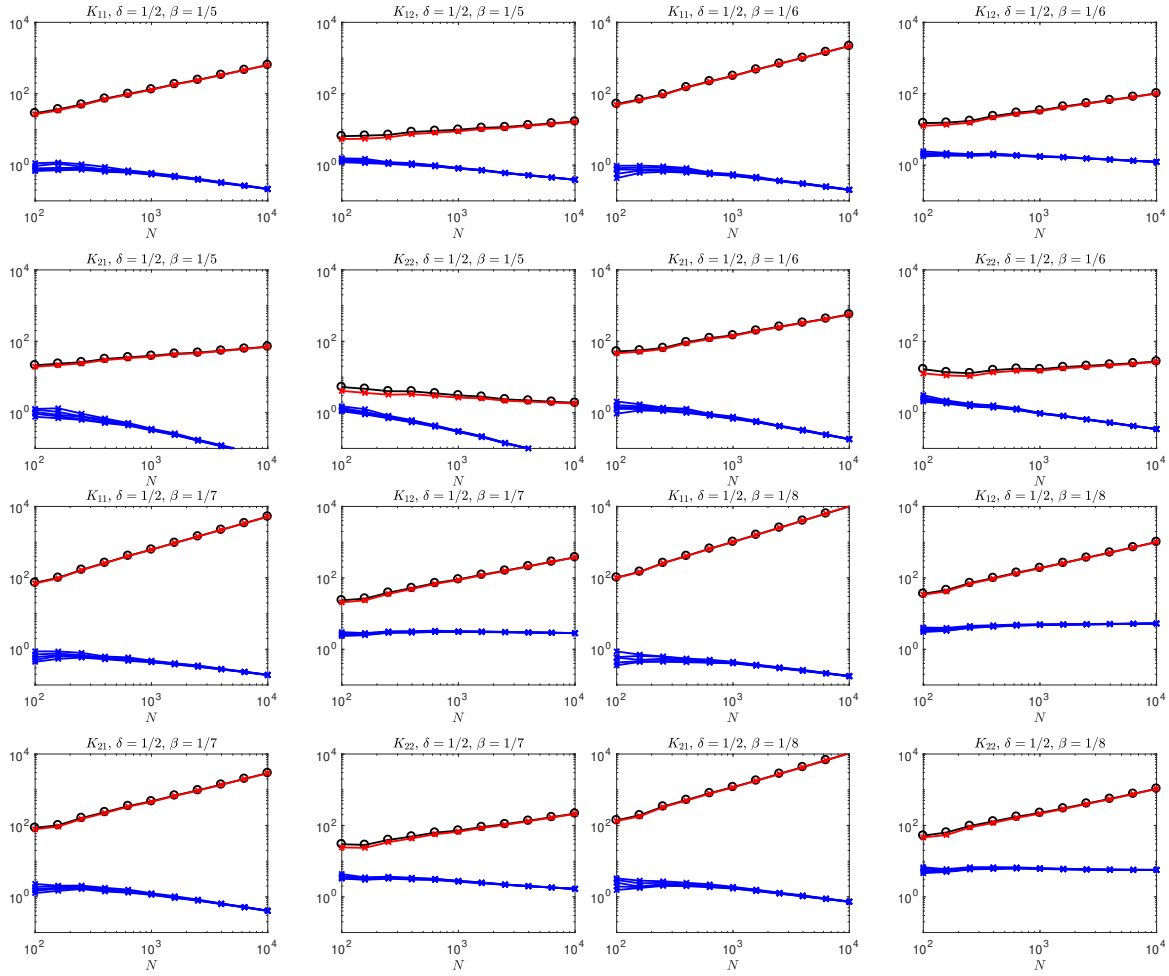

Figure 6: Same as in Figure 5 but with $\nu = O(N^{-\beta})$ for $\beta \in \{1/5, 1/6, 1/7, 1/8\}$.

that the indices $j, k, \ell, \iota = 1, \dots, K$ always denote *student* nodes, while $n, m = 1, \dots, M$ are reserved for teacher hidden nodes.

Rotating the dynamics The first step in the derivation is to rotate the order parameters into the basis given by the eigen-decomposition of the covariance matrix with eigenvalues ρ_τ and eigenvectors ψ_τ that are normalised as $\sum_\tau \psi_{\tau i}\psi_{\tau j} = N\delta_{ij}$ and $\sum_i \psi_{\tau i}\psi_{\tau' i} = N\delta_{\tau\tau'}$. We can then re-write the "teacher-student overlap" R (C.1) as

$$R^{km} = \frac{1}{\sqrt{\delta} N} \sum_\tau \Gamma_\tau^k \tilde{\Gamma}_\tau^m \tag{C.2}$$

where we have introduced the student and teacher projections

$$\Gamma_\tau^k \equiv \frac{1}{\sqrt{N}} \sum_i \psi_{\tau i} w_i^k, \qquad \tilde{\Gamma}_\tau^m \equiv \frac{1}{\sqrt{N}} \sum_i \psi_{\tau i} \tilde{\omega}_i^m, \qquad \tilde{\omega}_i^m \equiv \sum_r \Phi_{ir} \tilde{w}_r^m. \tag{C.3}$$

Note the normalisation (or lack thereof); this is due the fact that $\Phi_{ir} = \mathbb{E}\, x_i c_r \sim O(1/\sqrt{N})$. The student-student overlap becomes likewise

$$Q^{kl} = \frac{1}{N}\sum_{\tau} \rho_\tau \Gamma_\tau^k \Gamma_\tau^\ell, \tag{C.4}$$

and we also introduce a new teacher-teacher overlap, which is given by

$$\tilde{T}^{nm} = \frac{1}{N}\sum_{\tau} \tilde{\Gamma}_\tau^n \tilde{\Gamma}_\tau^m = \frac{1}{N}\sum_{i}\sum_{r,s} \tilde{w}_r^n \Phi_{ir}\Phi_{is}\tilde{w}_s^m \tag{C.5}$$

This order parameter can be interpreted as a teacher-teacher overlap with the teacher weights "rotated" by $\left[\Phi^\top \Phi\right]_{rs}$. This is a key observation: having the teacher act on the latent variables means that instead of having the actual teacher-teacher overlap, the student also sees a rotated version, rendering perfect learning impossible.

Teacher-student overlap To analyse quantities that are linear in the weights, such as the teacher-student overlap R^{km}, we have to analyse the SGD update

$$\mathrm{d}\Gamma_\tau^k = -\frac{\eta}{\sqrt{N}} v^k \left[\sum_{j\neq k}^{K} v^j \mathcal{A}_\tau^{jk} + v^k \mathcal{B}_\tau^k - \sum_{n}^{M} \tilde{v}^n \mathcal{C}_\tau^{nk}\right]. \tag{C.6}$$

We will use $\mathrm{d}$to denote the change in time-dependent quantities during one step of SGD. We have defined the following averages

$$\mathcal{A}_\tau^{jk} = \mathbb{E}\, g(\lambda^j) g'(\lambda^k)\beta_\tau, \qquad \mathcal{B}_\tau^{k} = \mathbb{E}\, g(\lambda^k) g'(\lambda^k)\beta_\tau, \qquad \mathcal{C}_\tau^{nk} = \mathbb{E}\, \tilde{g}(\nu^n) g'(\lambda^k)\beta_\tau. \tag{C.7}$$

where we have introduced the projected input

$$\beta_\tau \equiv \frac{1}{\sqrt{N}}\sum_i \psi_{\tau i} x_i. \tag{C.8}$$

As we discussed in the main text, there are now two crucial facts that make computing these averages possible. The online assumption asserts that at each step μ of SGD, the input x_μ used to evaluate the gradient is generated from a previously unused latent vector c_μ, which is uncorrelated to the students weights at that time. We also *assume* that the $K+M$ variables $\{\lambda^k, \nu^m\}$ are jointly Gaussian, making it possible to express the averages over $\{\lambda^k, \nu^m\}$ in terms of only their covariances, and hence later to close the equations. For the special-case of a single-layer generative network, Theorem 2 gives us verifiable conditions on the weights of the generator under which this holds. Using a simple Lemma 10 to evaluate the averages (C.7) yields

$$\begin{aligned}\mathcal{A}_\tau^{jk} = \frac{1}{Q^{kk}Q^{jj} - (Q^{kj})^2}\Big(&Q^{jj}\mathbb{E}\left[g'(\lambda^k)\lambda^k g(\lambda^j)\right]\ \mathbb{E}\left[\lambda^k \beta_\tau\right] - Q^{kj}\mathbb{E}\left[g'(\lambda^k)\lambda^j g(\lambda^j)\right]\ \mathbb{E}\left[\lambda^k\beta_\tau\right] \\ &- Q^{kj}\mathbb{E}\left[g'(\lambda^k)\lambda^k g(\lambda^j)\right]\ \mathbb{E}\left[\lambda^j\beta_\tau\right] + Q^{kk}\mathbb{E}\left[g'(\lambda^k)\lambda^j g(\lambda^j)\right]\ \mathbb{E}\left[\lambda^j\beta_\tau\right]\Big),\end{aligned} \tag{C.9}$$

and similarly for $\mathcal{B}_\tau^k$ and $\mathcal{C}_\tau^{nk}$. At this point, it is convenient to introduce a short-hand notation for the three-dimensional Gaussian averages

$$I_3(k,j,n) \equiv \mathbb{E}\left[g'(\lambda^k)\lambda^j \tilde{g}(\nu^n)\right], \tag{C.10}$$

which was introduced by Saad and Solla (1995a). Arguments passed to I_3 should be translated into local fields on the right-hand side by using the convention where the indices j, k, ℓ, ι always refer to student local fields λ^j, etc., while the indices n, m always refer to teacher local fields ν^n, ν^m. Similarly, $I_3(k, j, j) \equiv \mathbb{E}\left[g'(\lambda^k)\lambda^j g(\lambda^j)\right]$, where having the index j as the third argument means that the third factor is $g(\lambda^j)$, rather than $\tilde{g}(\nu^m)$ in Eq. (C.10). The average in Eq. (C.10) is taken over a three-dimensional normal distribution with mean zero and covariance matrix

$$\Phi^{(3)}(k, j, n) = \begin{pmatrix} Q^{kk} & Q^{kj} & R^{kn} \\ Q^{kj} & Q^{jj} & R^{jn} \\ R^{kn} & R^{jn} & T^{nn} \end{pmatrix}. \tag{C.11}$$

There are now two types of averages remaining. We first have $\mathbb{E}\,\lambda^k\beta_\tau = 1/\sqrt{N}\rho_\tau\Gamma^k_\tau$, and, likewise, $\mathbb{E}\,\nu^n\beta_\tau = 1/\sqrt{\delta N}\,\tilde{\Gamma}^n_\tau$. Putting everything together, we can write down the evolution of Γ^k_τ and identify the equations $h^{kj}_{(1)}$ etc. We have

$$\begin{aligned} \mathrm{d}\Gamma^k_\tau = -\frac{\eta}{N}v^k \Bigg(& \rho_\tau \sum_{j\neq k}\left[\Gamma^k_\tau v^j h^{kj}_{(1)}(Q) + v^j\Gamma^j_\tau h^{kj}_{(2)}(Q)\right] + \rho_\tau v^k\Gamma^k_\tau h^k_{(3)}(Q) \\ & - \sum_n \left[\rho_\tau \tilde{v}^n\Gamma^k_\tau h^{kn}_{(4)}(Q,R,T) + \frac{1}{\sqrt{\delta}}\tilde{v}^n\tilde{\Gamma}^n_\tau h^{kn}_{(5)}(Q,R,T)\right]\Bigg) \end{aligned} \tag{C.12}$$

where we have introduced the auxiliary functions $h^k_{(3)} = I_3(k,k,k)/Q^{kk}$ and

$$h^{kj}_{(1)} = \frac{Q^{jj}I_3(k,k,j) - Q^{kj}I_3(k,j,j)}{Q^{kk}Q^{jj} - (Q^{kj})^2} \qquad h^{kj}_{(2)} = \frac{Q^{kk}I_3(k,j,j) - Q^{kj}I_3(k,k,j)}{Q^{kk}Q^{jj} - (Q^{kj})^2} \tag{C.13a}$$

$$h^{kn}_{(4)} = \frac{T^{nn}I_3(k,k,n) - R^{kn}I_3(k,n,n)}{Q^{kk}T^{nn} - (R^{kn})^2} \qquad h^{kn}_{(5)} = \frac{Q^{kk}I_3(k,n,n) - R^{kn}I_3(k,k,n)}{Q^{kk}T^{nn} - (R^{kn})^2} \tag{C.13b}$$

Introducing order parameter densities We are now in a position to write down the equation for R^{km} Performing the sum over τ in Eq. (C.12), two types of terms remain. For the first four terms, we are left with the sum $\sum_\tau \rho_\tau\Gamma^k_\tau\tilde{\Gamma}^m_\tau$. This term cannot be reduced to an order parameter in a straightforward way. Instead, we can make progress by introducing the continuous function:

$$r^{km}(\rho) \equiv \frac{1}{\varepsilon_\rho}\frac{1}{N}\sum_\tau \Gamma^k_\tau\tilde{\Gamma}^m_\tau\ \mathbb{1}\left(\rho_\tau \in [\rho, \rho+\varepsilon_\rho[\right), \tag{C.14}$$

where $\mathbb{1}(\cdot)$ is the indicator function which evaluates to 1 if the condition given to it as an argument is true, and which otherwise evaluates to 0. We take the limit $\varepsilon_\rho \to 0$ after the thermodynamic limit. Then we can rewrite the order parameter R^{km} as an integral over the density r^{km}, weighted by the spectral density of the covariance Ω_{ij}:

$$R^{km} = \frac{1}{\sqrt{\delta}}\int \mathrm{d}\mu_\Omega(\rho)\ r^{km}(\rho). \tag{C.15}$$

For the final term in eq. (C.12), we introduce the density

$$\tilde{t}^{nm}(\rho) \equiv \frac{1}{\varepsilon_\rho}\frac{1}{N}\sum_\tau \tilde{\Gamma}^n_\tau\tilde{\Gamma}^m_\tau\ \mathbb{1}\left(\rho_\tau \in [\rho, \rho+\varepsilon_\rho[\right), \tag{C.16}$$

which allows us to write the first equation of motion, which we state in full in eq. (22).

Student-student overlap It is also convenient to re-write the student-student overlap as an integral

$$Q^{k\ell} = \int \mathrm{d}\mu_\Omega(\rho)\ \rho\ q^{k\ell}(\rho). \tag{C.17}$$

over a density $q^{kl}(\rho)$ that is defined analogously to $r^{km}(\rho)$,

$$q^{k\ell}(\rho) \equiv \frac{1}{\varepsilon_\rho}\frac{1}{N}\sum_\tau \Gamma_\tau^k \Gamma_\tau^\ell\ \mathbb{1}\left(\rho_\tau \in [\rho, \rho+\varepsilon_\rho[\right), \tag{C.18}$$

The part of the time-derivative of $q^{kl}(\rho)$ that is linear in Γ_τ can be obtained directly from eq. (C.12) as for R^{km}. For the quadratic part, we have to leading order in N

$$\frac{\eta^2}{N}\sum_\tau v^k v^\ell \mathbb{E}\,\Delta^2 g'(\lambda^k) g'(\lambda^\ell)\beta_\tau^2 = \eta^2\gamma v^k v^j \mathbb{E}\,\Delta^2 g'(\lambda^k) g'(\lambda^\ell) \tag{C.19}$$

where we used that $\mathbb{E}\,\beta_\tau^2 = \rho_\tau$ and we have defined $\gamma \equiv \sum_\tau \rho_\tau/N$, which is a constant of the motion. The remaining averages of the type $\mathbb{E}\,\Delta^2 g'(\lambda^k)g'(\lambda^\ell)$ can again be expressed succinctly using the shorthands Saad and Solla (1995a)

$$I_4(k,\ell,j,n) \equiv \mathbb{E}\left[g'(\lambda^k)g'(\lambda^\ell)g(\lambda^j)g(\nu^n)\right]. \tag{C.20}$$

that use the same notational conventions as for I_3. Putting it all together, we obtain the equation of motion (20) where we have introduced a final auxiliary function,

$$h^{k\ell}_{(6)}(Q,R,T,v,\tilde{v}) = \sum_{j,\iota}^{K} v^j v^\iota I_4(k,\ell,j,\iota) - 2\sum_{j}^{K}\sum_{m}^{M} v^j \tilde{v}^m I_4(k,\ell,j,m) + \sum_{n,m}^{M} \tilde{v}^n \tilde{v}^m I_4(k,\ell,n,m). \tag{C.21}$$

Second-layer weights Finally, we treat each of the second-layer weights of the student v as an order parameter in its own right. Their equations of motion (23) are readily found from from their SGD update (6)and require only the auxiliary funciton $h^{kn}_{(7)}(Q,R) \equiv \mathbb{E}\left[g(\lambda^k)g(\nu^n)\right]$ using the same convention for the subscript of $h^{kn}_{(7)}$ that we used for the integrals I_3 and I_4.

A simple lemma The derivation of the dynamical equations uses a simple Lemma that we recently used to analyse single-layer generators Goldt et al. (2020). To be as self-contained as possible, we repeat the Lemma here, and refer the interested reader to their paper for the proof.

Lemma 10 *Suppose you have T random variables $x^1, \dots, x^T$ with jointly Gaussian distribution $p(x^1, \dots, x^T)$. We assume that the distribution has zero first moments that the second moments matrix $q^{tt'}$ is positive definite. Suppose that an extra random variable y is jointly distributed with the $x^1, \dots, x^T$ and has mean zero, a finite variance $\langle y^2\rangle$, and correlations $\langle x^t y\rangle$ which are $O(1/\sqrt{N})$. Then for any two functions $\phi(x^1, \dots, x^T)$ and $\psi(y)$ that are odd in each of their arguments, we have, to leading order when $N \to \infty$:*

$$\langle \phi(x^1,\dots,x^T)\psi(y)\rangle = \sum_{t,s}(q^{-1})^{ts}\ \frac{\langle x^s y\rangle}{\langle y^2\rangle}\ \langle x^t \phi(x^1,\dots,x^T)\rangle\ \langle y\psi(y)\rangle \tag{C.22}$$

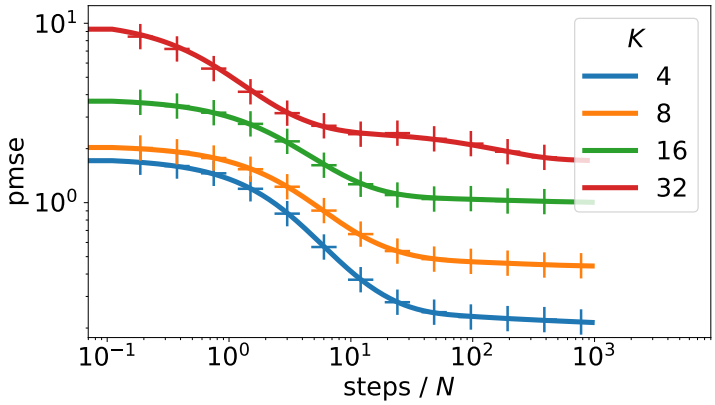

Figure 7: **Theory vs experiments for online SGD with increasingly large students**. We trained students with K hidden neurons on teachers with $M = K$ neurons with inputs coming from a single-layer generator (12) with random weights. $D = 500, N = 1000, \tilde{v}^m = 1, \eta = 0.05, g(x) = \tilde{g}(x) = \mathrm{erf}(x/\sqrt{2})$, integration time step $\mathrm{d}t = 0.01$.

C.1. Increasing the number of neurons

The dynamical equations we derived in this section are valid for any finite M, K after letting $N \to \infty$. For the simulations, it is thus natural to ask up to which number of neurons the equations accurately predict the dynamics for fixed N. We tested the accuracy of the equations by focusing on the single-layer generator (12) with $D = 500, N = 1000$. In this case, the Gaussian Equivalence holds rigorously thanks to Theorem 2, so as we increase M, K, we can expect deviations between theoretical predictions from the dynamical equations and simulations to arise only due to problems with the equations, rather than problems with Conjecture 1. We show the results of such an experiment in Fig. 7.

Appendix D. Replica analysis

In this Appendix we give the main steps in the replica derivation of the result in Section 3.2 for the full-batch learning. Our analysis, however, is restricted to the $K = M = 1$ case.

Setting: Consider the supervised learning problem introduced in Section 1 with $K = M = 1$. In this case, the model $y = \phi_\theta(\boldsymbol{x})$ is simply a *generalised linear model* with parameter $\boldsymbol{w} \in \mathbb{R}^N$:

$$\hat{y} = \phi_\theta(\boldsymbol{x}) = g\left(\frac{1}{\sqrt{N}}\boldsymbol{x}\cdot\boldsymbol{w}\right) \tag{D.1}$$

Similarly, we assume data in independently sampled $(\boldsymbol{x}, y) \sim q$ from the generative model introduced in eq. (2) with $M = 1$, which is equivalent to:

$$y = \phi_{\tilde{\theta}}(\boldsymbol{c}) = \tilde{g}\left(\frac{1}{\sqrt{D}}\boldsymbol{c}\cdot\tilde{\boldsymbol{w}}\right), \qquad \boldsymbol{x} = \mathcal{G}(\boldsymbol{c}), \qquad \boldsymbol{c} \sim \mathcal{N}(\boldsymbol{0}, \mathrm{I}_D) \tag{D.2}$$

where $\mathcal{G} : \mathbb{R}^D \to \mathbb{R}^N$ is a deep generative network as introduced in eq. (1), $\boldsymbol{c}$ is the latent variable and $\tilde{w} \sim P_{\tilde{w}}$ are a fixed set of weights. Different from the online analysis, here we are interested in characterising the generalisation performance of this model when trained on a batch of T independent samples from q. Let $\mathcal{D}_T = \{\boldsymbol{x}^\mu, y^\mu\}_{\mu=1}^T$ denote this training set. Training will consist on finding the set of weights $\hat{\boldsymbol{w}} \in \mathbb{R}^N$ that minimise the following empirical risk:

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}\in\mathbb{R}^N}{\operatorname{argmin}} \left[\sum_{\mu=1}^{T} \ell\left(y^\mu, \boldsymbol{x}^\mu \cdot \boldsymbol{w}\right) + \frac{\lambda}{2}||\boldsymbol{w}||_2^2 \right], \tag{D.3}$$

where ℓ is a generic loss function and we have added an ℓ_2 penalty with strength $\lambda > 0$. Our aim is to characterise the prediction error on a fresh set of samples $\boldsymbol{x}, y \sim q$,

$$\epsilon_g = \mathbb{E}_{(\boldsymbol{x},y)\sim q}\, \mathsf{pmse}(y, \hat{y}(\boldsymbol{x})), \tag{D.4}$$

in the high-dimensional limit where $N, P, D \to \infty$ while the ratios $\alpha = T/N$ (the sample complexity) and $\gamma = D/N$ (the compression rate) remain fixed. The key observation in our analysis is that precisely in this limit the asymptotic generalisation error can be fully characterised by only three scalar parameters $(\rho, m^\star, q^\star)$. Indeed, the *Gaussian Equivalence Property* (GEP) introduced in Section A allow us to write

$$\lim_{N\to\infty} \epsilon_g = \mathbb{E}_{\nu,\lambda}\left(\tilde{g}(\nu) - g(\lambda)\right)^2 \tag{D.5}$$

where $(\nu, \lambda) \sim \mathcal{N}(0, \Sigma)$ are jointly Gaussian random variables with covariance $\Sigma = \begin{pmatrix} \rho & m^\star \\ m^\star & q^\star \end{pmatrix}$ given by:

$$\rho = \frac{1}{D}||\tilde{\boldsymbol{w}}||_2^2, \qquad m^\star = \frac{1}{\sqrt{ND}}\hat{\boldsymbol{w}}^\top \Phi \tilde{\boldsymbol{w}}, \qquad q^\star = \frac{1}{N}\hat{\boldsymbol{w}}^\top \Omega \hat{\boldsymbol{w}} \tag{D.6}$$

with $\Phi = \mathbb{E}_{\boldsymbol{c}} \boldsymbol{x}\boldsymbol{c}^\top \in \mathbb{R}^{N\times D}$ and $\Omega = \mathbb{E}_{\boldsymbol{c}} \boldsymbol{x}\boldsymbol{x}^\top \in \mathbb{R}^{N\times N}$ being the *exact* covariances of the data. Note that ρ is completely fixed by $P_{\tilde{\boldsymbol{w}}}$. The replica analysis will give us $(m^\star, q^\star)$.

D.1. Replica analysis

The first step in the replica analysis is to define the following Gibbs measure over $\mathbb{R}^N$:

$$\mu_\beta(\boldsymbol{w}) = \frac{1}{\mathcal{Z}_\beta} e^{-\beta\left[\sum_{\mu=1}^{T} \ell(y^\mu, \boldsymbol{x}^\mu\cdot\boldsymbol{w}) + \frac{\lambda}{2}\sum_{i=1}^{N} w_i^2\right]} = \frac{1}{\mathcal{Z}_\beta} \underbrace{\prod_{\mu=1}^{T} e^{-\beta \sum_{\mu=1}^{T} \ell(y^\mu, \boldsymbol{x}^\mu\cdot\boldsymbol{w})}}_{P_y} \underbrace{\prod_{i=1}^{N} e^{-\frac{\beta\lambda}{2} w_i^2}}_{P_w} \tag{D.7}$$

where the normalisation $\mathcal{Z}_\beta$ is known as the *partition function*, and is a function of the training data $\mathcal{D}$. The factorised densities P_y and P_w can be interpreted as a (unormalised) likelihood and prior distribution respectively. Note that if we knew how to sample from μ_β, we would be able to solve eq. (D.3), since in the limit $\beta \to \infty$, the measure μ_β concentrates around solutions of this minimisation problem. The replica analysis consists in computing the *averaged free energy density*

$$\beta f_\beta = \lim_{N\to\infty} \frac{1}{N} \mathbb{E}_{\mathcal{D}} \log \mathcal{Z}_\beta \tag{D.8}$$

with the replica trick:

$$\log \mathcal{Z}_\beta = \lim_{r\to 0^+} \frac{1}{r}\partial_r \mathcal{Z}_\beta^r. \tag{D.9}$$

Linearising the logarithm allow us to average $\mathcal{Z}_\beta^r$ over the dataset explicitly. As we will see, once this average is taken, $\mathcal{Z}_\beta^r$ which is a priori a high-dimensional object (defined in terms of integrals in $\mathbb{R}^N$) factorise into a simple scalar quantities that will give us access to $(m^\star, q^\star)$.

Averaging over the data set: The average over the replicated partition function is explicitly given by:

$$\mathbb{E}_{\mathcal{D}}\mathcal{Z}_\beta^r = \prod_{\mu=1}^{T}\int \mathrm{d}y^\mu \int_{\mathbb{R}^D} \mathrm{d}\tilde{\boldsymbol{w}}\; P_{\tilde{w}}(\tilde{\boldsymbol{w}}) \int_{\mathbb{R}^{N\times r}} \left(\prod_{a=1}^{r} \mathrm{d}\boldsymbol{w}^a\; P_w(\boldsymbol{w}^a)\right) \times$$
$$\times \underbrace{\mathbb{E}_{\boldsymbol{c}^\mu}\left[\tilde{P}_y\left(y^\mu\middle|\frac{\boldsymbol{c}^\mu\cdot\tilde{\boldsymbol{w}}}{\sqrt{D}}\right)\prod_{a=1}^{r} P_y\left(y^\mu\middle|\frac{\boldsymbol{x}^\mu\cdot\boldsymbol{w}^a}{\sqrt{N}}\right)\right]}_{(\star)}$$

Note that since $\boldsymbol{x}^\mu = \mathcal{G}(\boldsymbol{c}^\mu)$ the average in $(\star)$ defines the joint probability between the random variables $\nu_\mu = \frac{\boldsymbol{c}^\mu\cdot\tilde{\boldsymbol{w}}}{\sqrt{D}}$ and $\lambda_\mu^a = \frac{\boldsymbol{x}^\mu\cdot\boldsymbol{w}^a}{\sqrt{N}}$. The *Gaussian Equivalence Principle* states that for certain architectures $\mathcal{G}$, the random variables (ν_μ, λ_μ^a) are asymptotically jointly Gaussian, with zero mean and covariance matrix given by:

$$\Sigma^{ab} = \begin{pmatrix} \rho & m^a \\ m^a & Q^{ab} \end{pmatrix}. \tag{D.10}$$

where the so-called overlap parameters (ρ, m^a, Q^{ab}) are related to the weights $\tilde{\boldsymbol{w}}, \boldsymbol{w}$:

$$\rho \equiv \mathbb{E}\left[\nu_\mu^2\right] = \frac{1}{D}||\tilde{\boldsymbol{w}}||_2^2, \quad m^a \equiv \mathbb{E}\left[\lambda_\mu^a\nu_\mu\right] = \frac{1}{\sqrt{ND}}\boldsymbol{w}^{a\top}\Phi\tilde{\boldsymbol{w}}, \quad Q^{ab} \equiv \mathbb{E}\left[\lambda_\mu^a\lambda_\mu^b\right] = \frac{1}{N}\boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^b$$

where all the information about the architecture of the generative network $\boldsymbol{x} = \mathcal{G}(\boldsymbol{c})$ is contained in the covariance matrices $\Omega = \mathbb{E}_{\boldsymbol{c}}\left[\boldsymbol{x}\boldsymbol{x}^\top\right]\mathbb{R}^{N\times N}$ and $\Phi = \mathbb{E}_{\boldsymbol{c}}\left[\boldsymbol{x}\boldsymbol{c}^\top\right] \in \mathbb{R}^{N\times D}$. We can therefore write the averaged replicated partition function as:

$$\mathbb{E}_{\mathcal{D}}\mathcal{Z}_\beta^r = \prod_{\mu=1}^{T}\int \mathrm{d}y^\mu \int_{\mathbb{R}^D} \mathrm{d}\tilde{\boldsymbol{w}}\; P_{\tilde{w}}(\tilde{\boldsymbol{w}}) \int_{\mathbb{R}^{N\times r}} \left(\prod_{a=1}^{r} \mathrm{d}\boldsymbol{w}^a\; P_w(\boldsymbol{w}^a)\right) \mathcal{N}(\nu_\mu, \lambda_\mu^a; \mathbf{0}, \Sigma^{ab}) \tag{D.11}$$

Rewriting as a saddle-point problem: The next step is to free the overlap parameters by introducing delta functions $\delta\left(D\rho - ||\tilde{\boldsymbol{w}}||_2^2\right)$, $\delta\left(\sqrt{ND}m^a - \boldsymbol{w}^a\Phi\tilde{\boldsymbol{w}}\right)$, $\delta\left(NQ^{ab} - \boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^b\right)$. Inserting in eq. (D.11), swapping the integrals and going to Fourier space allow us to rewrite:

$$\mathbb{E}_{\mathcal{D}}\mathcal{Z}_\beta^r = \int_{\mathbb{R}} \frac{\mathrm{d}\rho\,\mathrm{d}\hat{\rho}}{2\pi} \int_{\mathbb{R}^r} \prod_{a=1}^{r} \frac{\mathrm{d}m^a\,\mathrm{d}\hat{m}^a}{2\pi} \int_{\mathbb{R}^{r\times r}} \prod_{1\le a\le b\le r} \frac{\mathrm{d}Q^{ab}\,\mathrm{d}\hat{Q}^{ab}}{2\pi} e^{D\Phi^{(r)}} \tag{D.12}$$

where we have absorbed a $-i$ factor in the integrals[1] and defined the potential:

$$\Phi^{(r)} = -\gamma\rho\hat{\rho} - \sqrt{\gamma}\sum_{a=1}^{r} m^a\hat{m}^a - \sum_{1\leq a\leq b\leq r} Q^{ab}\hat{Q}^{ab} + \alpha\Psi_y^{(r)}(\rho, m^a, Q^{ab}) + \Psi_w^{(r)}(\hat{\rho}, \hat{m}^a, \hat{Q}^{ab})$$

with $\alpha = T/N$, $\gamma = D/N$ and:

$$\Psi_w^{(r)} = \frac{1}{N}\log\int_{\mathbb{R}^D} \mathrm{d}\tilde{\boldsymbol{w}} P_{w^\star}(\tilde{\boldsymbol{w}}) \int_{\mathbb{R}^{N\times r}} \prod_{a=1}^{r} \mathrm{d}\boldsymbol{w}^a P_w(\boldsymbol{w}^a)\, e^{\hat{\rho}||\tilde{\boldsymbol{w}}||_2^2 + \sum_{a=1}^{r}\hat{m}^a\boldsymbol{w}^{a\top}\Phi\tilde{\boldsymbol{w}} + \sum_{1\leq a\leq b\leq r}\hat{Q}^{ab}\boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^b}$$

$$\Psi_y^{(r)} = \log\int_{\mathbb{R}}\mathrm{d}y\int_{\mathbb{R}}\mathrm{d}\nu\,\tilde{P}_y(y|\nu)\int\prod_{a=1}^{r}\mathrm{d}\lambda^a P_y(y|\lambda^a)\,\mathcal{N}(\nu,\lambda^a;\mathbf{0},\Sigma^{ab}) \tag{D.13}$$

In the high-dimensional limit where $N\to\infty$ while $\alpha = T/N$ and $\gamma = D/N$ stay finite, the integral in eq. (D.12) concentrate around the values of the overlaps that extremise $\Phi^{(r)}$, and therefore we can write:

$$\beta f_\beta = -\lim_{r\to 0^+}\frac{1}{r}\mathrm{extr}\,\Phi^{(r)}\left(\hat{\rho}, \hat{m}^a, \hat{Q}^{ab}; \rho, m^a, Q^{ab}\right) \tag{D.14}$$

Replica symmetric ansatz: Finding the overlap configuration that minimise $\Phi^{(r)}$ is itself an intractable problem. In order to make progress, we restrict the extremisation above to the following *replica symmetric ansatz*:

$$\begin{aligned} m^a &= m, & \hat{m}^a &= \hat{m}, & &\text{for } a = 1,\dots,r \\ q^{aa} &= r, & \hat{q}^{aa} &= -\frac{1}{2}\hat{r}, & &\text{for } a = 1,\dots,r \\ Q^{ab} &= q, & \hat{Q}^{ab} &= \hat{q}, & &\text{for } 1\leq a<b\leq r \end{aligned} \tag{D.15}$$

Inserting this ansatz in eq. (D.13) allow us to explicitly take the $r\to 0^+$ limit for each term. The first three terms are trivial. The limit of $\Psi_y^{(r)}$ is cumbersome, but it common to many replica computations for the generalised linear likelihood P_y. We refer the curious reader to Gerace et al. (2020) for more details, and write the end result here:

$$\Psi_y \equiv \lim_{r\to 0^+}\frac{1}{r}\Psi_w^{(r)} = \mathbb{E}_\xi\left[\int_{\mathbb{R}}\mathrm{d}y\,\tilde{\mathcal{Z}}_y\left(y, \frac{m}{\sqrt{q}}\xi, \rho - \frac{m^2}{q}\right)\log\mathcal{Z}_y(y,\sqrt{q}\xi, V)\right] \tag{D.16}$$

where $\xi\sim\mathcal{N}(0,1)$, $V = r - q$ and:

$$\mathcal{Z}_y(y,\omega,V) = \int_{\mathbb{R}}\frac{\mathrm{d}x}{\sqrt{2\pi V}}e^{-\frac{(x-\omega)^2}{2V}}P_y(y|x), \quad \tilde{\mathcal{Z}}_y(y,\omega,V) = \int_{\mathbb{R}}\frac{\mathrm{d}x}{\sqrt{2\pi V}}e^{-\frac{(x-\omega)^2}{2V}}\tilde{P}_y(y|x) \tag{D.17}$$

Note that as in Gerace et al. (2020), the consistency condition of the zeroth order term in the free energy fix the parameters $\rho = \mathbb{E}_{P_{\tilde{w}}}\tilde{w}$ and $\hat{\rho} = 0$. The limit of $\Psi_w^{(r)}$ is slightly more involved. First, inserting the replica symmetric ansatz allow us to write:

$$\Psi_w^{(r)} = \frac{1}{N}\log\int_{\mathbb{R}^D}\mathrm{d}\tilde{\boldsymbol{w}}P_{w^\star}(\tilde{\boldsymbol{w}})\int_{\mathbb{R}^{N\times r}}\prod_{a=1}^{r}\mathrm{d}\boldsymbol{w}^a P_w(\boldsymbol{w}^a)\,e^{-\frac{\hat{V}}{2}\sum_{a=1}^{r}\boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^a + \hat{m}\sum_{a=1}^{r}\boldsymbol{w}^{a\top}\Phi\tilde{\boldsymbol{w}} + \hat{q}\sum_{a,b=1}^{r}\boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^b} \tag{D.18}$$

1. This won't matter since we will be only interested in the saddle-point of the integrals.

where we have defined $\hat{V} = \hat{r} + \hat{q}$. Now using that:

$$e^{\hat{q}\sum_{a,b=1}^{r} \boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^{b}} = \mathbb{E}_{\boldsymbol{\xi}}\left[e^{\sqrt{\hat{q}}\boldsymbol{\xi}^{\top}\Omega^{1/2}\sum_{a=1}^{r}\boldsymbol{w}^{a}}\right] \tag{D.19}$$

for $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathrm{I}_N)$, we can write:

$$\begin{aligned}\Psi_w^{(r)} &= \frac{1}{N}\log\int_{\mathbb{R}^D}\mathrm{d}\tilde{\boldsymbol{w}}P_{w^\star}(\tilde{\boldsymbol{w}})\prod_{a=1}^{r}\int_{\mathbb{R}^N}\mathrm{d}\boldsymbol{w}^a P_w(\boldsymbol{w}^a)\,\mathbb{E}_{\boldsymbol{\xi}}\left[e^{-\frac{\hat{V}}{2}\boldsymbol{w}^{a\top}\Omega\boldsymbol{w}^a+\boldsymbol{w}^{a\top}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\hat{q}\Omega^{1/2}\boldsymbol{\xi}\right)}\right]\\ &= \frac{1}{N}\log\mathbb{E}_{\boldsymbol{\xi}}\int_{\mathbb{R}^D}\mathrm{d}\tilde{\boldsymbol{w}}P_{w^\star}(\tilde{\boldsymbol{w}})\left[\int_{\mathbb{R}^N}\mathrm{d}\boldsymbol{w}\;P_w(\boldsymbol{w})\,e^{-\frac{\hat{V}}{2}\boldsymbol{w}^{\top}\Omega\boldsymbol{w}+\boldsymbol{w}^{\top}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\hat{q}\Omega^{1/2}\boldsymbol{\xi}\right)}\right]^r\end{aligned} \tag{D.20}$$

and therefore:

$$\Psi_w \equiv \lim_{r\to 0^+}\frac{1}{r}\Psi_w^{(r)} = \frac{1}{N}\mathbb{E}_{\boldsymbol{\xi},\tilde{\boldsymbol{w}}}\log\int_{\mathbb{R}^N}\mathrm{d}\boldsymbol{w}\;P_w(\boldsymbol{w})\,e^{-\frac{\hat{V}}{2}\boldsymbol{w}^{\top}\Omega\boldsymbol{w}+\boldsymbol{w}^{\top}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\hat{q}\Omega^{1/2}\boldsymbol{\xi}\right)} \tag{D.21}$$

Summary: The replica symmetric free energy density is simply given by:

$$\beta f_\beta = \underset{q,m,\hat{q},\hat{m}}{\mathrm{extr}}\left\{-\frac{1}{2}r\hat{r}-\frac{1}{2}q\hat{q}+m\hat{m}-\alpha\Psi_y(r,m,q)-\Psi_w(\hat{r},\hat{m},\hat{q})\right\} \tag{D.22}$$

where

$$\begin{aligned}\Psi_w &= \lim_{N\to\infty}\frac{1}{N}\mathbb{E}_{\xi,\tilde{\boldsymbol{w}}}\log\int_{\mathbb{R}^N}\mathrm{d}\boldsymbol{w}\;P_w(\boldsymbol{w})\,e^{-\frac{\hat{V}}{2}\boldsymbol{w}^{\top}\Omega\boldsymbol{w}+\boldsymbol{w}^{\top}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\hat{q}\Omega^{1/2}\boldsymbol{\xi}\right)}\\ \Psi_y &= \mathbb{E}_{\boldsymbol{\xi}}\left[\int_{\mathbb{R}}\mathrm{d}y\;\tilde{\mathcal{Z}}_y\left(y,\frac{m}{\sqrt{q}}\xi,\rho-\frac{m^2}{q}\right)\log\mathcal{Z}_y(y,\sqrt{q}\xi,V)\right]\end{aligned} \tag{D.23}$$

and

$$\mathcal{Z}_y(y,\omega,V) = \int_{\mathbb{R}}\frac{\mathrm{d}x}{\sqrt{2\pi V}}e^{-\frac{(x-\omega)^2}{2V}}P_y(y|x),\quad \tilde{\mathcal{Z}}_y(y,\omega,V) = \int_{\mathbb{R}}\frac{\mathrm{d}x}{\sqrt{2\pi V}}e^{-\frac{(x-\omega)^2}{2V}}\tilde{P}_y(y|x) \tag{D.24}$$

Simplifying Ψ_w: The result summarised above holds for any P_w and $P_{\tilde{w}}$, but can be considerably simplified in our case of interest eq. (D.7) where these densities are Gaussian. Indeed, we can integrate $\boldsymbol{w}$ explicitly in Ψ_w to get:

$$\begin{aligned}\int_{\mathbb{R}^N}\mathrm{d}\boldsymbol{w}\;P_w(\boldsymbol{w})e^{-\frac{\hat{V}}{2}\boldsymbol{w}^{\top}\Omega\boldsymbol{w}+\boldsymbol{w}^{\top}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\sqrt{\hat{q}}\Omega^{1/2}\boldsymbol{\xi}\right)} &= \int_{\mathbb{R}^N}\frac{\mathrm{d}\boldsymbol{w}}{(2\pi)^{p/2}}e^{-\frac{1}{2}\boldsymbol{w}^{\top}\left(\beta\lambda\mathrm{I}_N+\hat{V}\Omega\right)\boldsymbol{w}+\boldsymbol{w}^{\top}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\sqrt{\hat{q}}\Omega^{1/2}\boldsymbol{\xi}\right)}\\ &= \frac{\exp\left(\frac{1}{2}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\sqrt{\hat{q}}\Omega^{1/2}\boldsymbol{\xi}\right)^{\top}\left(\beta\lambda\mathrm{I}_N+\hat{V}\Omega\right)^{-1}\left(\hat{m}\Phi\tilde{\boldsymbol{w}}+\sqrt{\hat{q}}\Omega^{1/2}\boldsymbol{\xi}\right)^{\top}\right)}{\sqrt{\det\left(\beta\lambda\mathrm{I}_N+\hat{V}\Omega\right)}}\end{aligned} \tag{D.25}$$

where we have included a convenient rescaling of P_w. We can now take the log and average the resulting expression explicitly with respect to $P_{\tilde{w}} = \mathcal{N}(0,\mathrm{I}_N)$ and $\boldsymbol{\xi}\sim\mathcal{N}(0,\mathrm{I}_N)$. After some linear algebra manipulation, we can write the result (up to the limit) as:

$$\Psi_w = -\frac{1}{2N}\operatorname{tr}\log\left(\beta\lambda\mathrm{I}_N+\hat{V}\Omega\right)+\frac{1}{2N}\operatorname{tr}\left[\left(\hat{m}^2\Phi\Phi^{\top}+\hat{q}\Omega\right)\left(\beta\lambda\mathrm{I}_N+\hat{V}\Omega\right)^{-1}\right] \tag{D.26}$$

D.2. Saddle-point equations

In order to find the set of overlaps $(r^\star, \hat{r}^\star, q^\star, \hat{q}^\star, m^\star, \hat{m}^\star)$ that solve the extremisation problem in eq. (D.22), we look at the gradient of the replica symmetric potential. This give us a set of self-consistent equations known as *saddle-point equations*.

First, taking the gradient of Ψ_y with respect to (r, q, m) and recalling that $V = r - q$:

$$\partial_r \Psi_y = -\mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y \frac{\partial_\omega \mathcal{Z}_y^2}{\mathcal{Z}_y} \right], \quad \partial_q \Psi_y \quad = \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y f_y^2 \right], \quad \partial_m \Psi_y = \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \partial_\omega \tilde{\mathcal{Z}}_y f_y \right]$$

where $f_y \equiv \log \mathcal{Z}_y$. Now looking at the gradient of Ψ_w with respect to $(\hat{r}, \hat{q}, \hat{m})$ and recalling that $\hat{V} = \hat{r} + \hat{q}$:

$$\begin{aligned}
\partial_{\hat{r}} \Psi_w &= -\frac{1}{2N} \operatorname{tr} \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-1} \Omega - \frac{\hat{m}^2}{2N} \operatorname{tr} \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-2} \Omega\Phi\Phi^\top - \frac{\hat{q}}{2N} \operatorname{tr} \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-2} \Omega^2 \\
\partial_{\hat{q}} \Psi_w &= -\frac{\hat{m}^2}{2N} \operatorname{tr} \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-2} \Omega\Phi\Phi^\top - \frac{\hat{q}}{2N} \operatorname{tr} \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-2} \Omega^2 \\
\partial_{\hat{m}} \Psi_w &= \frac{\hat{m}}{d} \operatorname{tr} \Phi^\top \Phi \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-1}
\end{aligned} \tag{D.27}$$

Putting together give the following set of self-consistent saddle-point equations:

$$\begin{cases} \hat{V} = \alpha \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y \partial_\omega f_y \right] \\ \hat{q} = \alpha \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y f_y^2 \right] \\ \hat{m} = \frac{\alpha}{\sqrt{\gamma}} \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \partial_\omega \tilde{\mathcal{Z}}_y f_y \right] \end{cases} \qquad \begin{cases} V = \frac{1}{N} \operatorname{tr} \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-1} \Omega \\ q = \frac{1}{N} \operatorname{tr} \left[\left(\hat{q}\Omega + \hat{m}^2 \Phi\Phi^\top \right) \Omega \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-2} \right] \\ m = \frac{\hat{m}}{N\sqrt{\gamma}} \operatorname{tr} \Phi\Phi^\top \left(\beta\lambda \mathrm{I}_N + \hat{V}\Omega \right)^{-1} \end{cases} \tag{D.28}$$

where we used $\partial_\omega f_y = \mathcal{Z}_y^{-1} \partial_\omega^2 \mathcal{Z} - f_y^2$. To take the $\beta \to \infty$ limit explicitly, we look at the following ansatz for the scaling of the order parameters:

$$\begin{aligned}
V^\infty &= \beta V & q^\infty &= q & m^\infty &= m \\
\hat{V}^\infty &= \frac{1}{\beta} \hat{V} & \hat{q}^\infty &= \frac{1}{\beta^2} \hat{q} & \hat{m}^\infty &= \frac{1}{\beta} \hat{m}.
\end{aligned} \tag{D.29}$$

With this scaling, we can easily get rid of the β dependency in the equations for (V, q, m). For the $(\hat{V}, \hat{q}, \hat{m})$ equations, we note that:

$$\mathcal{Z}_y(y, \sqrt{q}\xi, V) = \int \frac{\mathrm{d}x}{\sqrt{2\pi V}} e^{-\frac{(x - \sqrt{q}\xi)^2}{2V}} e^{-\beta \ell(y,x)} = \int \frac{\mathrm{d}x}{\sqrt{2\pi V}} e^{-\beta \left[\frac{(x - \sqrt{q^\infty}\xi)^2}{2V^\infty} + \ell(y,x) \right]} \tag{D.30}$$

and therefore when $\beta \to \infty$, $\mathcal{Z}_y$ is dominated by the exponential of the values that minimise the argument in the exponent, which is the *proximal operator* associated to the loss ℓ:

$$\eta(y, \omega, V) = \underset{x \in \mathbb{R}}{\operatorname{argmin}} \left[\frac{(x - \omega)^2}{2V} + \ell\left(y, x\right) \right] \tag{D.31}$$

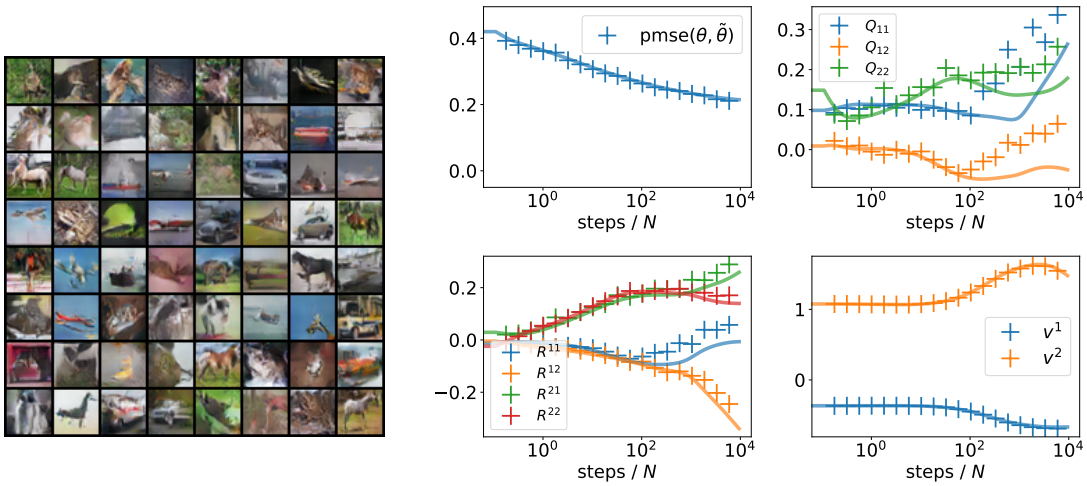

Figure 8: **Theory vs experiments for online SGD with deep, pre-trained dcGAN of Radford et al. (2016).** *(Left)* The top four rows show images drawn randomly from the CIFAR10 data set, the bottom four rows show images drawn randomly from the pre-trained dcGAN. *(Right)* Same plot as Fig. 2 when inputs are drawn from the pre-trained realNVP. $D = N = 3072, M = K = 2, \tilde{v}^m = 1, \eta = 0.2, g(x) = \tilde{g}(x) = \mathrm{erf}(x/\sqrt{2})$, integration time step $\mathrm{d}t = 0.01$.

Finally, in the $\beta \to \infty$ limit the saddle-point equations can be written as:

$$\begin{cases} \hat{V} = \alpha \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y \left(\frac{1 - \partial_\omega \eta}{V} \right) \right] \\ \hat{q} = \alpha \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \tilde{\mathcal{Z}}_y \left(\frac{\eta - \omega}{V} \right)^2 \right] \\ \hat{m} = \frac{\alpha}{\sqrt{\gamma}} \mathbb{E}_\xi \left[\int_{\mathbb{R}} \mathrm{d}y \; \partial_\omega \tilde{\mathcal{Z}}_y \left(\frac{\eta - \omega}{V} \right) \right] \end{cases} \qquad \begin{cases} V = \frac{1}{N} \mathrm{tr} \left(\lambda \mathrm{I}_N + \hat{V} \Omega \right)^{-1} \Omega \\ q = \frac{1}{N} \mathrm{tr} \left[\left(\hat{q} \Omega + \hat{m}^2 \Phi \Phi^\top \right) \Omega \left(\lambda \mathrm{I}_N + \hat{V} \Omega \right)^{-2} \right] \\ m = \frac{\hat{m}}{N \sqrt{\gamma}} \mathrm{tr} \, \Phi \Phi^\top \left(\lambda \mathrm{I}_N + \hat{V} \Omega \right)^{-1} \end{cases} \tag{D.32}$$

where we have dropped the $\cdot^\infty$ superscript to lighten the notation. This is the expression quoted on the main text. Note that for convex loss functions, the problem in eq. (D.3) is strongly convex, and therefore admit one and only one solution $\hat{w}$. This implies that the solution for the overlaps $(m^\star, q^\star)$ found by iterating the saddle-point equations above *necessarily* coincides with the overlaps appearing in the expression for the generalisation error given by eq. (D.5). This means that the replica symmetric fully characterises the generalisation performance in the convex case.

Appendix E. Further experimental results

Results for online SGD with the pre-trained dcGAN We also compared the dynamical equations to simulations in the case of the dcGAN pre-trained on CIFAR10 images, see Fig 8. We see that in this case, the equations capture the evolution of the pmse well and exactly predict the evolution of the second-layer weights v. This is a crucial result, since we obtain these predictions from analytical

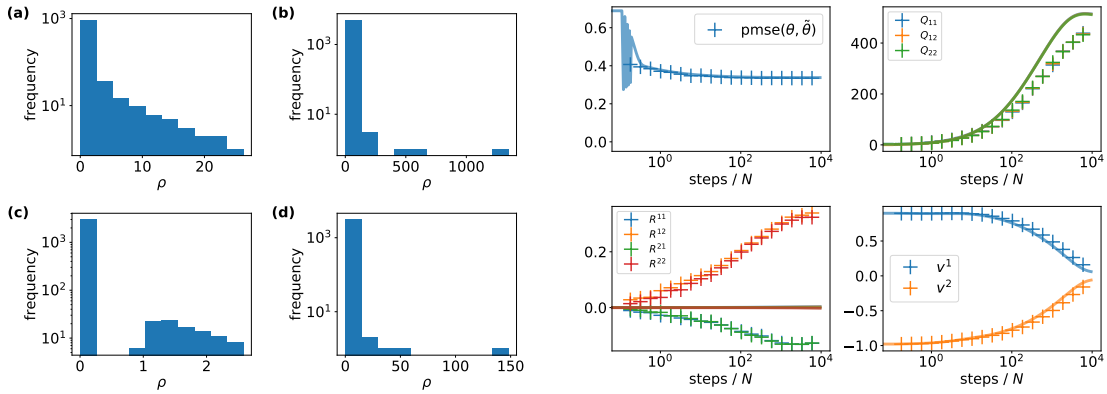

Figure 9: **The impact of the spectral density of the input-input covariance on learning.** *(Left)*: Spectral density of the average covariance matrix of inputs drawn from four generative models: (a) Random fully-connected network of Fig. 2, (b) fully connected generator with inverse weights (see Sec. E), (c) dcGAN with random weights, and (d) dcGAN trained on CIFAR10. *(Right):* We compare theory vs simulation for the training of two-layer neural network on inputs x drawn from a two-layer, fully connected generative network where the weights of the second layer are the matrix inverse of the first layer, Eq. (E.1). $D = 5000, N = 5000, M = K = 2, \tilde{v}^m = 1, \eta = 0.2, g(x) = \tilde{g}(x) = \mathrm{erf}(x/\sqrt{2})$, integration time step $\mathrm{d}t = 0.01$.

expressions for the functions $h^{kn}_{(7)}$ and $h^{kj}_{(8)}$ that are only valid if the GEP holds. One can therefore interpret the correct predictions for v based on the GEP as experimental evidence that the GEP holds for this pre-trained convolutional generators. The results for the order parameters Q and R reveal larger fluctuations after about $100N \sim 10^5$ SGD steps, for example for Q^{11} (blue line in top right plot). One source of error here is numerical and due to the small size of the teacher network $(D = 100)$ to which we are comparing a theory that holds asymptotically, i.e. when $N, D \to \infty$. Such a small teacher would lead to deviations from the ODEs due to finite-size effects even for i.i.d. Gaussian inputs. To confirm that these deviations are finite-size effects, we also verified our theory for a different class of generative model, the aforementioned normalising flows, who have a larger latent dimension D. As we see in Sec. 4.3, the ODEs perfectly agree with simulations for this model with larger input dimension.

Generative model with strongly correlated weights Finally, we also constructed a generative model with strongly correlated weights where there exists a dominant direction in the eigenspace of the input-input covariance matrix $\Omega_{ij} = \mathbb{E}\, x_i x_j$. We took a fully connected generative network $\mathcal{G} : \mathbb{R}^N \to \mathbb{R}^N$, with two layers of weights $A^1 \in \mathbb{R}^{N\times N}$ and $A^2 \in \mathbb{R}^{N\times N}$. We drew the elements of A^1 element-wise i.i.d. from the standard normal distribution, whereas the second-layer weights $A^2 = \mathrm{inv}(A^1)$. After each layer, we used the sign activation function, so the generator's output function can be written as

$$x = \mathcal{G}(c) = \mathrm{sign}\left(\mathrm{inv}(A^1)\mathrm{sign}\left(A^1 c\right)\right) \tag{E.1}$$

On the left of Fig. 9, we show the spectra of the covariance matrices of various generators. The leading eigenvalues are smallest for generators with random weights, such as the fully-connected single-layer network (12) (a) and the dcGAN with random weights (c) that we used in Fig. 2. The pre-trained dcGAN has a leading eigenvalue that is about an order of magnitude larger (d). The generator with inverse weights (E.1) has an eigenvalue that is yet another order of magnitude larger.

The particular weight structure of the "inverse" generator also has a strong impact on the dynamics of a two-layer network trained on its data, as we show on the right of Fig. 9. Notably, the length of the weight vectors grows exponentially for a large portion of training time, while the second-layer weights go to zero. We observed this behaviour consistently over several runs of this setup with different weights for the teacher, generator and different initial weights for the student in each case. Characterising the impact of a dominant direction in the data on the dynamics of two-layer neural networks is an intriguing challenge that we leave for future work.