



Quantitative Gaussian approximation of randomly initialized deep neural networks

Andrea Basteri^{1,2,3} · Dario Trevisan⁴

Received: 10 February 2023 / Revised: 24 April 2024 / Accepted: 30 May 2024 /

Published online: 25 June 2024

© The Author(s), under exclusive licence to Springer Science+Business Media LLC, part of Springer Nature 2024

Abstract

Given any deep fully connected neural network, initialized with random Gaussian parameters, we bound from above the quadratic Wasserstein distance between its output distribution and a suitable Gaussian process. Our explicit inequalities indicate how the hidden and output layers sizes affect the Gaussian behaviour of the network and quantitatively recover the distributional convergence results in the wide limit, i.e., if all the hidden layers sizes become large.

Keywords Neural networks · Central limit theorem · Wasserstein distance

Mathematics Subject Classification 60F05 · 60G15 · 68T99

1 Introduction

Contemporary machine learning has seen a surge in applications of deep neural networks, achieving state-of-the-art performances in many contexts, from speech and visual recognition (LeCun et al., 2015), to feature extraction and generation of samples (Goodfellow et al., 2014). Such a deep learning revolution (Sejnowski, 2018) is often compared to the advent of the industrial age (Roberts et al. 2022, Chap. 0), to the extent that the search for powerful and efficient steam machines historically led to the first theoretical investigations

Editor: Hendrik Blockeel

Member of the INdAM GNAMPA group.

✉ Andrea Basteri
andrea.basteri@inria.fr

Dario Trevisan
dario.trevisan@unipi.it

¹ INRIA Paris, 75012 Paris, France

² Département d'Informatique, École Normale Supérieure, Paris, France

³ PSL Research University, Paris, France

⁴ Dipartimento di Matematica, Università di Pisa, Largo Bruno Pontecorvo 5, 56127 Pisa, Italy

in thermodynamics. This analogy may be a bit of a stretch, but the fundamental questions about the scope and limits of deep learning methods have indeed stimulated new mathematical research in various fields, most notably probability, statistics and statistical physics, with exciting recent progresses towards enlightening answers.

1.1 Random neural networks

A successful approach focuses on the scaling limit of large neural networks whose parameters are randomly sampled. Besides the introduction of several mathematical tools, there are convincing reasons for such a study. Firstly, a Bayesian approach to machine learning problems would require a prior distribution on the set of networks, which should then be updated via Bayes rule, given the observations, e.g., the training set in supervised learning. In practice, however, large neural networks are trained via iterative optimisation algorithms (Goodfellow et al., 2016), which still require a careful initialization of the parameters, often suggested to be randomly sampled (Narkhede et al., 2022), thus providing a second motivation for such a study. Even more strikingly, training only the parameters in the last layer of a randomly initialized neural network may still provide good performances and of course an extreme speed-up in the training algorithm, a fact that stimulated a whole machine learning paradigm (Cao et al., 2018).

In view of the above reasons, the study of random neural networks is thus not new, and in fact dates back to pioneers in the field (Rosenblatt, 1958). In the seminal work (Neal, 1996), motivated by the Bayesian approach, Neal first proved that wide shallow networks, i.e., with a large number of parameters but only one hidden layer, converge to Gaussian processes, if the parameters are random and suitably tuned, as a consequence of the Central Limit Theorem (CLT). More recently, in de Matthews et al. (2018), Lee et al. (2018), convergence has been extended to deeper networks, stemming a series of works that aim to explain the success story of over-parametrized neural networks, totally unexpected from traditional statistical viewpoints, by linking it to the well-established theory of Gaussian processes (Williams & Rasmussen, 2006). A breakthrough is the realization (Lee et al., 2019) that the training dynamics, e.g. via gradient descent, can be also traced in the scaling limit and become a differential equation of gradient flow type associated to the loss functional and the so-called Neural Tangent Kernel, i.e., a quadratic form associated to the gradient with respect to the parameters of the network, evaluated on the training set. Let us also mention the alternative perspective of the so-called mean field limit of networks (Mei et al., 2018; Nguyen & Pham, 2023), where parameters are tuned in a Law of Large Numbers regime, leading to deterministic non-linear dynamics for the network training.

1.2 Our contribution

In this work, we provide a quantitative proof of the Gaussian behaviour of deep neural networks with random parameters at initialization, i.e., without addressing the correlations induced by the training procedure. We thus complement the seminal works (de G. Matthews et al., 2018) and subsequent ones focusing on convergence for wide networks (Yang, 2019; Bracale et al., 2021; Hanin, 2023), by providing for the first time explicit rates for convergence in distribution in the case of deep networks. The metric we adopt is the Wasserstein distance, which generalizes the so-called earth mover's distance, and provide a natural interpretation of the discrepancy between the two distributions in terms of the average cost between transporting one over the other. Indeed, it was

historically introduced by Monge as the Optimal Transport problem, i.e., to efficiently move a given distribution of mass from a location to another (Villani, 2009). The theory of optimal transport has grown in the last decades, with applications beyond calculus of variation and probability, from geometry and partial differential equations (Ambrosio et al., 2005), to statistics and machine learning (Peyré et al., 2019). We use in fact only basic properties of the Wasserstein distance, that are all recalled in Sect. 2.2, and focus for simplicity of the exposition only on the quadratic case, denoted by $\mathcal{W}_2$, i.e., when the transport cost between two locations is the squared Euclidean distance.

To further keep the exposition short and accessible, we discuss only so-called fully connected neural networks (or multi-layer perceptrons), where all units (neurons) in each layer are linked with those on the next layer, from the input layer to the output (see Fig. 1). We denote by $L + 1$ the total number of layers (including input and output), with sizes n_0 for the input layer, n_L for the output and $(n_1, \dots, n_{L-1})$ for the intermediate hidden ones. Then, for every choice of the parameters, often called weights (matrices) $\mathbf{W} = (W^{(\ell)})_{\ell=0}^{L-1}$ and biases (vectors) $\mathbf{b} = (b^{(\ell)})_{\ell=0}^{L-1}$, and a (usually fixed) non-linear activation function σ acting on each intermediate node (except for the last layer), a neural network $f^{(L)} : \mathbb{R}^{n_0} \rightarrow \mathbb{R}^{n_L}$ is defined (see Sect. 2.3 for a precise definition).

Using the concepts and notation mentioned above, our main result can be stated as follows (see Theorem 2 for a more precise version).

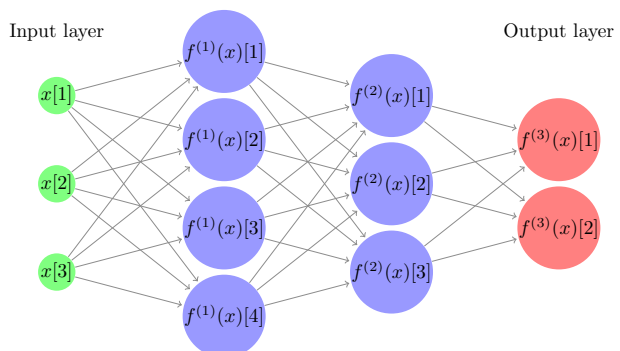
Theorem 1 Consider a fully connected neural network $f^{(L)} : \mathbb{R}^{n_0} \rightarrow \mathbb{R}^{n_L}$ with Lipschitz activation function σ , and random weights $\mathbf{W}$ and biases $\mathbf{b}$ that are independent Gaussian random variables, centered with

$$\mathbb{E}[(W_{ij}^{(\ell)})^2] = \frac{1}{n_\ell}, \quad \mathbb{E}[(b_i^{(\ell)})^2] = 1, \quad \text{for every } \ell = 0, 1, \dots, L-1 \text{ and all } i, j. \quad (1)$$

Then, for every given set of k inputs $\mathcal{X} = \{x_i\}_{i=1}^k \subseteq \mathbb{R}^{n_0}$, the distribution of $f^{(L)}[\mathcal{X}] = (f^{(L)}(x_i))_{i=1}^k$, i.e., the law of the output of the random network evaluated at the inputs $\mathcal{X}$, is quantitatively close to a centered multivariate Gaussian $\mathcal{N}(K^{(L)}[\mathcal{X}])$ with a (computable) covariance matrix $K^{(L)}[\mathcal{X}]$, in terms of the quadratic Wasserstein distance:

$$\mathcal{W}_2(\mathbb{P}_{f^{(L)}[\mathcal{X}]}, \mathcal{N}(K^{(L)}[\mathcal{X}])) \leq C \sqrt{n_L} \sum_{\ell=1}^{L-1} \frac{1}{\sqrt{n_\ell}}, \quad (2)$$

Fig. 1 Graphical representation (Cowan, 2022) of a fully connected neural network with $L = 3$ layers, input size $n_0 = 3$, output size $n_3 = 2$ and hidden layer sizes $n_1 = 4, n_2 = 3$



where $C \in (0, \infty)$ is a constant depending on the activation function σ , the input $\mathcal{X}$ and the number of layers L only, not on the hidden or output layer sizes $(n_\ell)_{\ell=1}^L$.

The random network considered in our result is also known as the Xavier initialization, which sets the initial weights using a scaling factor that takes into account the number of input and output neurons in each layer. Let us point out that several other random distributions can be considered for specific activation functions or network architectures, see (Narkhede et al., 2022) for a review.

The matrix $K^{(L)}[\mathcal{X}]$ depends on the activation function σ , the input $\mathcal{X}$ and the output dimension n_L (not on the hidden layer sizes $(n_\ell)_{\ell=1}^{L-1}$). It is indeed recursively computable (see Sect. 2.4) and in some cases – e.g. for ReLU activation – closed formulas can be obtained, see (Cho & Saul, 2009; Lee et al., 2018). An important feature is that all n_L output neurons in the Gaussian approximation are independent and identically distributed variables (for any input) – the desirability of such property is often discussed, but let us recall that we are considering here non-trained networks.

A useful feature of our result is that it explicitly indicates how the layer sizes affect the Gaussian behaviour of the output of the network. In particular, in the wide limit where all the hidden layers sizes n_i 's are large (keeping n_L and n_0 fixed) the Wasserstein distance becomes infinitesimal and convergence in distribution for fixed input set follows. Some considerations apply as well in the narrow deep limit, since the constant C in (2) can be made explicit and also more general variances in (1) can be introduced: the contribution $\sqrt{n_L}/\sqrt{n_\ell}$ naturally associated to the ℓ -th hidden layer is weighted by exponentially contracting (or growing) terms as L becomes large (see Remark 1).

1.3 Related literature

Our proof is based on the natural idea, already present in Neal (1996), that the Gaussian limit emerges due to a combination, in each layer, of the CLT scaling for the weights parameters and the inherited convergence in the previous layer, which almost yields independence for the neurons. For shallow networks, which correspond to the base case of our argument, exact independence holds and Gaussian behaviour is immediate from the CLT. When considering deeper architectures, the generalization is not immediate and de Mathews et al. (2018) made use of a CLT for triangular arrays. The arguments were later simplified, e.g. via the use of characteristic functions (Bracale et al., 2021) or moments (Hanin, 2023), but quantitative convergence in terms of any natural distance (not only the Wasserstein one) was missing so far. It turns out in fact that using the Wasserstein distance allows for quantitatively tracking the hidden layers and yields a relatively straightforward proof by induction, where the triangle inequality is employed to obtain exact independence from the previous layers, and most terms are estimated without much effort.

Let us point out that our main result provides convergence for any finite set of k -inputs of a random deep neural network. Obtaining rates for deep networks in suitable functional spaces (informally, for infinitely many inputs) seems considerably more difficult, although useful, e.g. to study architectures with bottlenecks (Agrawal et al., 2020), i.e., when some hidden layers that are purposely kept with small width, such as in autoencoders. The case of shallow networks has been addressed in Eldan et al. (2021), Klukowski (2022), using Stein's method for the quadratic Wasserstein distance in the space of squared integrable functions on the sphere. In Sect. 5 we hint at how one could further extend our argument to obtain bounds in functional spaces.

1.4 Extensions and future work

As already mentioned, we choose to keep technicalities at minimum by considering only the quadratic Wasserstein distance, fully connected architectures, Gaussian weights and biases and a finite subset of inputs. In view of the existing literature on the subject, we expect however that all such aspects may be relaxed, e.g., from Wasserstein distances of any order p or further variants, such as the entropic regularized optimal transport (Peyré et al., 2019); to more general network architectures, such as those of convolutional type (Borovykh, 2019; Garriga-Alonso et al., 2019), following e.g. the general tensor program formulation proposed in Yang (2019), Yang and Littwin (2021); to non-Gaussian laws for the random parameters, such as discrete cases or heavy-tailed distributions like stable laws where the Gaussian CLT fails (Peluchetti et al., 2020), by using finer probabilistic tools, such as quantitative multi-dimensional CLT's in the Wasserstein space (Bonis, 2020).

Another interesting question is whether (2) are sharp in some sense (eventually allowing for general non Gaussian distributions in the weights and biases parameters) and what are the properties, e.g. regularity, of the optimal transport map pushing the Gaussian law to the distribution of the output of the neural network, whose existence follows from general results (Villani, 2009, theorem 9.4): more precise information may help to transfer finer results from Gaussian processes to neural networks.

A further direction concerns the validity of similar quantitative approximations after the training procedure, e.g. by coupling it with the gradient flow description provided by the already mentioned Neural Tangent Kernel theory. Alternatively, one could investigate if the arguments in Eldan et al. (2021) for the exact Bayes posterior may be combined with our quantitative bounds.

1.5 Structure of the paper

Section 2 is devoted to introduce notation and basic facts about neural networks, Gaussian processes and the Wasserstein distance. Section 3 contains a more precise statement of our main result, Theorem 2 and its entire proof. Section 4 gives with the proof of technical Lemma 4, while Sect. 5 contains some remarks on the functional limit (i.e., for infinite input sets). Finally, in Sect. 6 we report the outcomes of some numerical experiments about the sharpness of our main result.

2 Notation

Given a set S , we write $\mathbb{R}^S$ for the set of real valued functions $f : S \rightarrow \mathbb{R}$, and write $s \mapsto f(s)$, but also frequently use the notation $f[s] = f_s$, to avoid subscripts (in vectors and matrices). For a subset $T \subseteq S$, we denote by $f[T] = (f[t])_{t \in T} \in \mathbb{R}^T$ its restriction to T . When $S = \{1, 2, \dots, k\}$, then we simply write $\mathbb{R}^S = \mathbb{R}^k$, which is the set of k -dimensional vectors $v = (v[i])_{i=1}^k$, while if $S = \{1, \dots, h\} \times \{1, \dots, k\}$ then we write $\mathbb{R}^S = \mathbb{R}^{h \times k}$, the set of matrices $A = (A[i, j])_{i=1, \dots, h}^{j=1, \dots, k}$. The space $\mathbb{R}^S$ enjoys a natural structure of vector space with

pointwise sum $(f + g)[s] = f[s] + g[s]$ and product $(\lambda f)[s] = \lambda f[s]$ and a canonical basis $(e_s)_{s \in S}$ with $e_s[t] = 1$ if $s = t$ and $e_s[t] = 0$ otherwise.

Given sets S, T , the tensor product operation between $f \in \mathbb{R}^S$ and $g \in \mathbb{R}^T$ defines $f \otimes g \in \mathbb{R}^{S \times T}$ with $(f \otimes g)[s, t] = f(s)g(t)$. The canonical basis for $\mathbb{R}^{S \times T}$ can be then written as $(e_{(s,t)}) = e_s \otimes e_t$, $s \in S, t \in T$. If S, T are finite, then any $A \in \mathbb{R}^{S \times T}$ can be identified also with the space of linear transformations from $\mathbb{R}^T$ to $\mathbb{R}^S$ via the usual matrix product

$$A : f \mapsto Af \quad \text{where} \quad (Af)[s] = \sum_{t \in T} A[s, t]f[t].$$

The identity matrix $\text{Id}_S \in \mathbb{R}^{S \times S}$ is defined as $\text{Id}_S = \sum_s e_s \otimes e_s$, i.e., $\text{Id}_S[s, t] = e_s[t]$, and write Id_k or $\text{Id}_{h \times k}$ in the case of $\mathbb{R}^k$ or $\mathbb{R}^{h \times k}$ respectively. If $S = S_1 \times S_2$ and $T = T_1 \times T_2$ and $A \in \mathbb{R}^{S_1 \times T_1}$, $B \in \mathbb{R}^{S_2 \times T_2}$, then, up to viewing $A \otimes B \in \mathbb{R}^{(S_1 \times T_1) \times (S_2 \times T_2)}$ as an element in $\mathbb{R}^{(S_1 \times S_2) \times (T_1 \times T_2)}$, the following identity holds:

$$(A \otimes B)(f \otimes g) = (Af) \otimes (Bg)$$

for every $f \in \mathbb{R}^{T_1}$, $g \in \mathbb{R}^{T_2}$. When $S = T$, then any $A \in \mathbb{R}^{S \times S}$ can be also seen as a quadratic form over $\mathbb{R}^S$: A is symmetric if A equals its transpose $A^T[s, t] = A[t, s]$, i.e., $A[s, t] = A[t, s]$ for $s, t \in S$ and positive semidefinite if, for every finite subset $T \subseteq S$ the restriction of A to $T \times T$, that we denote by $A[T]$ (instead of $A[T \times T]$) is positive semidefinite as a matrix, i.e., for every $v \in \mathbb{R}^T$,

$$\sum_{s, t \in T} v[s]A[s, t]v[t] \geq 0.$$

If S is finite, we define the trace of $A \in \mathbb{R}^{S \times S}$ as $\text{tr}(A) = \sum_{s \in S} A[s, s]$. We use this notion to define a scalar product on $\mathbb{R}^S$ as

$$\langle f, g \rangle = \text{tr}(f \otimes g) = \sum_{s \in S} f[s]g[s],$$

which induces the norm

$$\|f\| = \sqrt{\text{tr}(f \otimes f)} = \sqrt{\sum_{s \in S} |f[s]|^2},$$

yielding the Euclidean norm in case of a vectors in $\mathbb{R}^k$ and the Frobenius (or Hilbert-Schmidt) norm for matrices $\mathbb{R}^{h \times k}$. The tensor product is associative and the norm is multiplicative, i.e., $\|f \otimes g\| = \|f\|\|g\|$.

Given a symmetric positive semi-definite $A \in \mathbb{R}^{S \times S}$ with S finite, its square root $\sqrt{A}$ is defined via spectral theorem as $\sqrt{A} = U\sqrt{D}U^T$ where $U \in \mathbb{R}^{S \times S}$ is orthogonal, i.e., $UU^T = \text{Id}$ and $U^T A U = D = \sum_s d_s e_s \otimes e_s$ is diagonal with $d_s \geq 0$ (the eigenvalues of A) and $\sqrt{D} = \sum_s \sqrt{d_s} e_s \otimes e_s$ is the diagonal with entries being the square roots of the entries of D . If $A, B \in \mathbb{R}^{S \times S}$ are symmetric and positive semi-definite, then (Hemmen and Ando, 1980, Proposition 3.2) gives

$$\|\sqrt{A} - \sqrt{B}\| \leq \frac{1}{\sqrt{\lambda(A)}} \|A - B\|, \quad (3)$$

where $\lambda(A) = \min_s d_s$ denotes its smallest eigenvalue (possibly zero: in such a case we interpret the right hand side as $+\infty$). Finally, we will use the fact that $\sqrt{A \otimes B} = \sqrt{A} \otimes \sqrt{B}$ if $A \in \mathbb{R}^{S \times S}$, $B \in \mathbb{R}^{T \times T}$ are both symmetric and positive-semidefinite, by interpreting the identity in $\mathbb{R}^{(S \times T) \times (S \times T)}$.

2.1 Random variables and Gaussian processes

For a random variable X with values on $\mathbb{R}^S$, write $\mathbb{E}[X] \in \mathbb{R}^S$ for its mean value, i.e., $\mathbb{E}[X][s] = \mathbb{E}[X[s]]$. The second moment of X is defined as $\mathbb{E}[X \otimes X] \in \mathbb{R}^{S \times S}$, i.e., the collection $(\mathbb{E}[X[s]X[t]])_{s,t \in S}$, which is symmetric and positive semi-definite. We always tacitly assume that these are well-defined. The covariance matrix $\Sigma(X) \in \mathbb{R}^{S \times S}$ of X is defined as the second moment of the centered variable $X - \mathbb{E}[X]$, which coincides with its second moment if X is centered, i.e., $\mathbb{E}[X] = 0$. Recalling that $\|X\|^2 = \text{tr}(X \otimes X)$, and exchanging expectation and the trace operation, we have that $\mathbb{E}[\|X\|^2] = \text{tr}(\mathbb{E}[X \otimes X])$, hence

$$\mathbb{E}[\|X\|^2] = \text{tr}(\Sigma(X)), \quad \text{if } X \text{ is centered.} \quad (4)$$

If A is a random variable with values in $\mathbb{R}^{T \times S}$, its second moment

$$\mathbb{E}[A \otimes A] \in \mathbb{R}^{(T \times S) \times (T \times S)}$$

can be viewed also as in $\mathbb{R}^{(T \times T) \times (S \times S)}$, which is a useful point of view we often adopt.

Given a symmetric positive semi-definite $K \in \mathbb{R}^{S \times S}$, we write $\mathcal{N}(K)$ for the centered Gaussian distribution on $\mathbb{R}^S$ with covariance (or, equivalently, second moment) K , i.e., the law of any Gaussian random variable X with values in $\mathbb{R}^S$, $\mathbb{E}[X] = 0$ and $\Sigma(X) = K$. If S is infinite, the law is Gaussian if for every finite subset $T \subseteq S$, the restriction $X[T] = (X[t])_{t \in T}$ has a multivariate Gaussian law with covariance $K[T] = K[T \times T]$. The variable $X = (X[s])_{s \in S}$ is also called a Gaussian process (or field) over S .

2.2 Wasserstein distance

We recall without proof basic facts about the Wasserstein distance, referring to any monograph such as Villani (2009), Ambrosio et al. (2005), Peyré et al. (2019), Santambrogio (2015) for details. Let S be finite. The quadratic Wasserstein distance between two probability measures p, q on $\mathbb{R}^S$ is defined as

$$\mathcal{W}_2(p, q) = \inf \left\{ \sqrt{\mathbb{E}[\|X - Y\|^2]} : X, Y \text{ random variables with } \mathbb{P}_X = p, \mathbb{P}_Y = q \right\},$$

where the infimum runs over all the random variables X, Y (jointly defined on some probability space) with marginal laws $p = \mathbb{P}_X$ and $q = \mathbb{P}_Y$. For simplicity, we allow for a slight abuse of notation and often write $\mathcal{W}_2(X, Y)$ instead of $\mathcal{W}_2(\mathbb{P}_X, \mathbb{P}_Y)$, when we consider two random variables X, Y with values on $\mathbb{R}^S$. In particular, it holds

$$\mathcal{W}_2(X, Y) \leq \sqrt{\mathbb{E}[\|X - Y\|^2]}. \quad (5)$$

With this notation, a sequence of random variables $(X_n)_n$ converges towards a random variable X , i.e., $\lim_{n \rightarrow \infty} \mathcal{W}_2(X_n, X) = 0$ if and only if $\lim_{n \rightarrow \infty} X_n$ in law together with their

second moments $\lim_{n \rightarrow \infty} \mathbb{E}[X_n \otimes X_n] = \mathbb{E}[X \otimes X]$. Given X, Y, Z all with values on $\mathbb{R}^S$, the triangle inequality holds

$$\mathcal{W}_2(X, Z) \leq \mathcal{W}_2(X, Y) + \mathcal{W}_2(Y, Z), \quad (6)$$

and, if Z is independent of X and Y , the following inequality holds:

$$\mathcal{W}_2(X + Z, Y + Z) \leq \mathcal{W}_2(X, Y). \quad (7)$$

The distance squared enjoys a useful convexity property, which we can state as follows: given random variables X, Y , with values on $\mathbb{R}^S$ and Z , taking values on some $\mathbb{R}^T$, then

$$\mathcal{W}_2^2(X, Y) \leq \int_{\mathbb{R}^T} \mathcal{W}_2^2(\mathbb{P}_{X|Z=z}, \mathbb{P}_Y) d\mathbb{P}_Z(z). \quad (8)$$

Finally, if X, Y are centered Gaussian random variables with values on $\mathbb{R}^S$ (S finite), it holds

$$\mathcal{W}_2(X, Y) \leq \left\| \sqrt{\Sigma(X)} - \sqrt{\Sigma(Y)} \right\|, \quad (9)$$

which can be easily seen by considering a standard Gaussian variable Z on $\mathbb{R}^S$, i.e., centered with $\Sigma(Z) = \text{Id}_S$ and letting $X = \sqrt{\Sigma(X)}Z$ and $Y = \sqrt{\Sigma(Y)}Z$.

2.3 Deep neural networks

We consider only the so-called fully connected (or multi-layer perceptron) architecture, which consists of a sequence of hidden layers stacked between the input and output layers, where each node is connected with all nodes in the subsequent layer. Let $L \geq 1$ and let $\mathbf{n} = (n_0, n_1, \dots, n_L)$ denote the size of the layers, so that the input belongs to $\mathbb{R}^{n_0}$ and the overall network outputs a vector in $\mathbb{R}^{n_L}$. Given an activation function $\sigma : \mathbb{R} \rightarrow \mathbb{R}$ and a set of weights and biases (in this section they do not need to be random variables)

$$\mathbf{W} = (W^{(0)}, W^{(1)}, \dots, W^{(L-1)}), \quad \mathbf{b} = (b^{(0)}, b^{(1)}, \dots, b^{(L-1)}),$$

where, for every $\ell = 0, \dots, L-1$, one has $W^{(\ell)} \in \mathbb{R}^{n_{\ell+1} \times n_\ell}$ and $b^{(\ell)} \in \mathbb{R}^{n_{\ell+1}}$, we define

$$f^{(1)} : \mathbb{R}^{n_0} \rightarrow \mathbb{R}^{n_1}, \quad f^{(1)}[x] = W^{(0)}x + b^{(0)},$$

and recursively, for $\ell = 2, \dots, L$,

$$f^{(\ell)} : \mathbb{R}^{n_0} \rightarrow \mathbb{R}^{n_\ell}, \quad f^{(\ell)}[x] = W^{(\ell-1)}\sigma(f^{(\ell-1)}[x]) + b^{(\ell-1)},$$

where the composition σ will be always understood componentwise. Notice that $f^{(\ell)}$ is a function of weights and biases $(W^{(i)}, b^{(i)})_{i=0}^{\ell-1}$. Given a finite subset of inputs $\mathcal{X} = \{x_i\}_{i=1, \dots, k} \subseteq \mathbb{R}^{n_0}$, we identify it with a slight abuse of notation with a matrix $\mathcal{X} = \sum_{i=1}^k x_i \otimes e_i \in \mathbb{R}^{n_0 \times k}$ and consider the restrictions $f^{(\ell)}[\mathcal{X}]$ that we interpret also as matrices

$$f^{(\ell)}[\mathcal{X}] = \sum_{i=1}^k f^{(\ell)}[x_i] \otimes e_i \in \mathbb{R}^{n_\ell \times k}.$$

We notice that

$$f^{(1)}[\mathcal{X}] = (W^{(0)} \otimes \text{Id}_k) \mathcal{X} + b^{(0)} \otimes 1_k, \quad (10)$$

$$f^{(\ell)}[\mathcal{X}] = (W^{(\ell-1)} \otimes \text{Id}_k) \sigma(f^{(\ell-1)}[\mathcal{X}]) + b^{(\ell-1)} \otimes 1_k, \quad (11)$$

for $\ell = 2, \dots, L$, where $1_k = \sum_{i=1}^k e_i \in \mathbb{R}^k$ denotes the vector with all entries equal to 1.

2.4 Neural network Gaussian processes

The Gaussian process associated to a wide random neural network (Lee et al., 2018) is defined merely in terms of its covariance operator (it is centered) and ultimately depends only on the choice of the activation function σ , the non-negative weights and biases variances

$$\left(c_w^{(0)}, c_b^{(0)}, c_w^{(1)}, c_b^{(1)}, \dots, c_w^{(L-1)}, c_b^{(L-1)} \right) \in \mathbb{R}_+^{2L}$$

as well as the layer dimensions. As with the construction of the network, the definitions are recursive. We first define the kernel $K^{(1)} = (K^{(1)}[x, y])_{x, y \in \mathbb{R}^{n_0}}$ as

$$K^{(1)}[x, y] = \frac{c_w^{(0)}}{n_0} \langle x, y \rangle + c_b^{(0)} = \frac{c_w^{(0)}}{n_0} \sum_{i=1}^{n_0} x[i]y[i] + c_b^{(0)}.$$

It is not difficult to check that $K^{(1)}$ is symmetric and positive semi-definite, thus there exists a centered Gaussian process $(G^{(1)}[x])_{x \in \mathbb{R}^{n_0}}$ on $\mathbb{R}^{n_0}$ with covariance kernel $K^{(1)}$, i.e.,

$$\mathbb{E}[G^{(1)}[x]G^{(1)}[y]] = K^{(1)}[x, y].$$

For $\ell = 2, \dots, L$, having already defined the positive semi-definite kernel $K^{(\ell-1)} = (K^{(\ell-1)}[x, y])_{x, y \in \mathbb{R}^{n_{\ell-1}}}$, we denote with $(G^{(\ell-1)}[x])_{x \in \mathbb{R}^{n_{\ell-1}}}$ a centered Gaussian process with covariance kernel $K^{(\ell-1)}$ and use it to define

$$K^{(\ell)}[x, y] = c_w^{(\ell-1)} \mathbb{E}[\sigma(G^{(\ell-1)}[x])\sigma(G^{(\ell-1)}[y])] + c_b^{(\ell-1)},$$

which can be easily seen to be positive semi-definite. For later use, let us also introduce the kernel

$$K_0^{(\ell)}[x, y] = c_w^{(\ell-1)} \mathbb{E}[\sigma(G^{(\ell-1)}[x])\sigma(G^{(\ell-1)}[y])]. \quad (12)$$

Finally, given layer dimensions $\mathbf{n} = (n_0, n_1, \dots, n_L)$, the Gaussian process approximating the layer output $f^{(\ell)}[x]$, is defined as n_ℓ independent copies of $G^{(\ell)}$, i.e., a Gaussian process on $S = \mathbb{R}^{n_0} \times \{1, \dots, n_\ell\}$, whose covariance kernel is then $K^{(\ell)} \otimes \text{Id}_{n_\ell}$. To keep notation simple, at the cost of a slight ambiguity of the notation, we also write $G^{(\ell)}[x] = (G^{(\ell)}[x, i])_{i=1, \dots, n_\ell} \in \mathbb{R}^{n_\ell}$ for the (vector-valued) Gaussian process. In analogy with the notation for neural networks, given k inputs $\mathcal{X} = \{x_i\}_{i=1}^k$, write then $G^{(\ell)}[\mathcal{X}] = \sum_{i=1}^k G^{(\ell)}[x_i] \otimes e_i \in \mathbb{R}^{n_\ell \times k}$, which is a centered Gaussian random variable with covariance

$$\Sigma(G^{(\ell)}[\mathcal{X}]) = \text{Id}_{n_\ell} \otimes K^{(\ell)}[\mathcal{X}] \in \mathbb{R}^{(n_\ell \times n_\ell) \times (k \times k)},$$

where $K^{(\ell)}[\mathcal{X}] = (K^{(\ell)}[x, y])_{x, y \in \mathcal{X}}$.

3 Proof of main result

We consider the fully connected architecture described in the previous section, where weights $\mathbf{W}$ and biases $\mathbf{b}$ are all independent Gaussian random variables, centered with variances

$$\Sigma(W^{(\ell)}) = \frac{c_w^{(\ell)}}{n_\ell} \text{Id}_{n_{\ell+1} \times n_\ell}, \quad \Sigma(b^{(\ell)}) = c_b^{(\ell)} \text{Id}_{n_{\ell+1}},$$

where $(c_w^{(\ell)}, c_b^{(\ell)})_{\ell=0}^{L-1} \in \mathbb{R}_+^{2L}$. We also assume that the activation function σ is Lipschitz, i.e., for some finite constant $\text{Lip}(\sigma)$

$$|\sigma(z) - \sigma(z')| \leq \text{Lip}(\sigma) |z - z'| \quad \text{for all } z, z' \in \mathbb{R}. \quad (13)$$

This is the case with $\text{Lip}(\sigma) = 1$ for the common choice of ReLU activation $\sigma(z) = z^+$.

□

We state and prove a more precise version of our main result.

Theorem 2 *With the above notation for the deep neural network outputs $\{f^{(\ell)}[\mathcal{X}]\}_{\ell=1}^L$ with random weights and biases, and the associated Gaussian processes $\{G^{(\ell)}[\mathcal{X}]\}_{\ell=2}^L$ evaluated at k inputs $\mathcal{X} = \{x_i\}_{i=1}^k \subseteq \mathbb{R}^{n_0}$, there exists positive finite constants $(C^{(\ell)})_{\ell=1}^L$ such that, for every $\ell = 1, \dots, L$, it holds*

$$\mathcal{W}_2(f^{(\ell)}[\mathcal{X}], G^{(\ell)}[\mathcal{X}]) \leq \sqrt{n_\ell} \sum_{i=1}^{\ell-1} \frac{C^{(i+1)} \text{Lip}(\sigma)^{\ell-1-i} \sqrt{\prod_{j=i+1}^{\ell-1} c_w^{(j)}}}{\sqrt{n_i}}. \quad (14)$$

Each $C^{(\ell)}$ depends upon σ , $\mathcal{X}$ and only the weights $(c_w^{(i)}, c_b^{(i)})_{i=0}^\ell$.

Remark 1 Inequality (14) is evidently more precise than (2). In particular, besides recovering the Gaussian convergence of the network outputs (for a finite input set) in the wide limit, i.e., if all $(n_i)_{i=1}^{L-1}$ become large, it allows for some partial consideration in the deep limit, i.e., if L becomes large and all the n_i 's are uniformly bounded. Let $\ell = L \rightarrow \infty$ in (14) and assume e.g. that $\text{Lip}(\sigma) \leq 1$ (as in the ReLU case) and all weights variances are constants, $c_w^{(i)} = c_w$. The i -th term in the sum is then multiplied by the exponential factor $\sqrt{c_w^{L-i-2}}$ which becomes negligible as $L \rightarrow \infty$, if $c_w < 1$. According to Roberts et al. (2022, Sect. 3.4), the “critical” case in the deep narrow limit is indeed the choice $c_w = 1$, where the network behaviour is “chaotic” (and not Gaussian), and indeed our inequality becomes ineffective.

Before we provide the proof of Theorem 2, which is performed by induction over $\ell = 1, \dots, L$, we inform the readers that we extensively use tensor product notation. This is necessary to take into account the joint distributions of the various outputs, which are independent only in the Gaussian case (or in the wide network limit). The interested

reader that would prefer to avoid tensor products but still have a good understanding of the argument is strongly invited to specialize our derivation to the case of a single input and a one-dimensional output.

Base case. The case $\ell = 1$ is straightforward, since $f^{(1)}[\mathcal{X}]$ is a linear function of the Gaussian variable $W^{(0)}$ and $b^{(0)}$, thus it has Gaussian law, centered with covariance

$$\begin{aligned}\Sigma(f^{(1)}[\mathcal{X}]) &= \Sigma((W^{(0)} \otimes \text{Id}_k)\mathcal{X} + b^{(0)} \otimes 1_k) \\ &= \Sigma((W^{(0)} \otimes \text{Id}_k)\mathcal{X}) + \Sigma(b^{(0)} \otimes 1_k) \quad \text{by independence,} \\ &= \text{Id}_{n_1} \otimes K^{(1)}[\mathcal{X}, \mathcal{X}],\end{aligned}$$

where in the last step we used the following computation which we state in a general form for later use (here we used $A = \mathcal{X}$ and $\ell = 0$).

Lemma 3 For any $\ell = 0, \dots, L-1$, given $A \in \mathbb{R}^{n_\ell \times k}$, one has

$$\Sigma((W^{(\ell)} \otimes \text{Id}_k)A) = \text{Id}_{n_{\ell+1} \times n_{\ell+1}} \otimes \bar{A}, \quad (15)$$

where $\bar{A}[j_1, j_2] = \frac{c^{(\ell)}}{n_\ell} \sum_{m=1}^{n_\ell} A[m, j_1]A[m, j_2]$ and

$$\mathbb{E}\left[\left\|(W^{(\ell)} \otimes \text{Id}_k)A\right\|^2\right] = \frac{c^{(\ell)}n_{\ell+1}}{n_\ell} \|A\|^2. \quad (16)$$

Proof The first identity is an explicit computation, using the covariance of $W^{(\ell)}$:

$$\begin{aligned}\Sigma((W^{(\ell)} \otimes \text{Id}_k)A)[i_1, j_1, i_2, j_2] &= \mathbb{E}\left[(W^{(\ell)} \otimes \text{Id}_k)A[i_1, j_1](W^{(\ell)} \otimes \text{Id}_k)A[i_2, j_2]\right] \\ &= \sum_{m_1, m_2=1}^{n_\ell} \mathbb{E}\left[W^{(\ell)}[i_1, m_1]A[m_1, j_1]W^{(\ell)}[i_2, m_2]A[m_2, j_2]\right] \\ &= e_{i_1}[i_2] \frac{c^{(\ell)}}{n_\ell} \sum_{m=1}^{n_\ell} A[m, j_1]A[m, j_2].\end{aligned}$$

The second identity follows from (4). $\square$

Induction step. We assume that (14) holds for some $1 \leq \ell \leq L-1$ and prove it for $\ell+1$. To simplify the notation, we omit to write $[\mathcal{X}]$ in what follows, and consider any probability space where random variables with the same laws as $f^{(\ell)}[\mathcal{X}]$ and $G^{(\ell)}[\mathcal{X}]$ are jointly defined (but we assume nothing on their joint law). We denote them as $f^{(\ell)}$ and $G^{(\ell)}$.

Without loss of generality, we assume that the weights $W^{(\ell)}$ and biases $b^{(\ell)}$ are defined on the same space and independent of $f^{(\ell)}$ and $G^{(\ell)}$, otherwise we enlarge it to define independent copies of $W^{(\ell)}$ and $b^{(\ell)}$. We define auxiliary random variables

$$h^{(\ell+1)} = (W^{(\ell)} \otimes \text{Id}_k)\sigma(G^{(\ell)}), \quad g^{(\ell+1)} = h^{(\ell+1)} + b^{(\ell)} \otimes 1_k.$$

To take into account the biases, which play practically no role in our bounds, we also assume that a centered Gaussian random variable $H^{(\ell+1)}$ is defined on the same space, with values in $\mathbb{R}^{n_{\ell+1} \times k}$ and covariance

$$\text{Id}_{n_{\ell+1}} \otimes K_0^{(\ell+1)}[\mathcal{X} \times \mathcal{X}] \in \mathbb{R}^{(n_{\ell+1} \times n_{\ell+1}) \times (k \times k)},$$

with the kernel $K_0^{(\ell+1)}$ introduced in (12). Without loss of generality, we also assume that $H^{(\ell+1)}$ is independent from $b^{(\ell)}$, so that

$$H^{(\ell+1)} + b^{(\ell)} \otimes 1_k$$

is Gaussian with the same law as $G^{(\ell+1)} = G^{(\ell+1)}[\mathcal{X}]$.

By the triangle inequality, we now split

$$\mathcal{W}_2(f^{(\ell+1)}, G^{(\ell+1)}) \leq \mathcal{W}_2(f^{(\ell+1)}, g^{(\ell+1)}) + \mathcal{W}_2(g^{(\ell+1)}, G^{(\ell+1)}),$$

and bound separately the two terms.

First term. We prove that

$$\mathcal{W}_2^2(f^{(\ell+1)}, g^{(\ell+1)}) \leq \frac{n_{\ell+1}}{n_{\ell}} c^{(\ell)} \text{Lip}(\sigma)^2 \mathbb{E} \left[\|f^{(\ell)} - G^{(\ell)}\|^2 \right]. \quad (17)$$

Indeed,

$$\begin{aligned} \mathcal{W}_2^2(f^{(\ell+1)}, g^{(\ell+1)}) &= \mathcal{W}_2^2(W^{(\ell)}\sigma(f^{(\ell)}) + b^{(\ell)}, W^{(\ell)}\sigma(G^{(\ell)}) + b^{(\ell)}) \\ &\leq \mathcal{W}_2^2(W^{(\ell)}\sigma(f^{(\ell)}), W^{(\ell)}\sigma(G^{(\ell)})) \quad \text{by (7),} \\ &\leq \mathbb{E} \left[\|W^{(\ell)}\sigma(f^{(\ell)}) - W^{(\ell)}\sigma(G^{(\ell)})\|^2 \right] \quad \text{by (5).} \end{aligned}$$

To proceed further, we condition upon $f^{(\ell)}$ and $G^{(\ell)}$, so that, with $A = \sigma(f^{(\ell)}) - \sigma(G^{(\ell)})$ in (16), we obtain

$$\mathbb{E} \left[\|W^{(\ell)}\sigma(f^{(\ell)}) - W^{(\ell)}\sigma(G^{(\ell)})\|^2 \mid f^{(\ell)}, G^{(\ell)} \right] = \frac{c^{(\ell)} n_{\ell+1}}{n_{\ell}} \|\sigma(f^{(\ell)}) - \sigma(G^{(\ell)})\|^2.$$

Finally, we use (13), to bound from above

$$\|\sigma(f^{(\ell)}) - \sigma(G^{(\ell)})\|^2 \leq \text{Lip}(\sigma)^2 \|f^{(\ell)} - G^{(\ell)}\|^2,$$

and conclude with (17).

Second term. We prove that

$$\mathcal{W}_2^2(f^{(\ell+1)}, G^{(\ell+1)}) \leq \frac{n_{\ell+1}}{n_{\ell}} C^{(\ell+1)}, \quad (18)$$

for a positive finite constant $C^{(\ell+1)}$ depending on $\mathcal{X}$, σ and $(c_w^{(i)}, c_b^{(i)})_{i=0}^{\ell}$ only (an explicit expression is (19) below).

To this aim, we use again (7) to remove the bias terms,

$$\begin{aligned} \mathcal{W}_2(g^{(\ell+1)}, G^{(\ell+1)}) &= \mathcal{W}_2(h^{(\ell+1)} + b^{(\ell)} \otimes 1_k, H^{(\ell+1)} + b^{(\ell)} \otimes 1_k) \\ &\leq \mathcal{W}_2(h^{(\ell+1)}, H^{(\ell+1)}). \end{aligned}$$

Next, we use (8) with $X = h^{(\ell+1)}$, $Y = H^{(\ell+1)}$ and $Z = G^{(\ell)}$. Indeed, conditioning upon $G^{(\ell)} = z \in \mathbb{R}^{n_{\ell} \times k}$, the random variable

$$h^{(\ell+1)} = (W^{(\ell)} \otimes \text{Id}_k) \sigma(G^{(\ell)}) = (W^{(\ell)} \otimes \text{Id}_k) \sigma(z)$$

has centered Gaussian law whose covariance is provided by (15) applied with $A = \sigma(z)$: $\text{Id}_{n_{\ell+1}} \otimes \overline{\sigma(z)}$, where $\overline{\sigma(z)} \in \mathbb{R}^{k \times k}$ is given by

$$\overline{\sigma(z)}[j_1, j_2] = \frac{c_w^{(\ell)}}{n_{\ell}} \sum_{m=1}^{n_{\ell}} \sigma(z[m, j_1]) \otimes \sigma(z[m, j_2]).$$

By (9), we bound from above

$$\begin{aligned} \mathcal{W}_2^2(\mathbb{P}_{h^{(\ell+1)}|G^{(\ell)}=z}, \mathbb{P}_{H^{(\ell+1)}}) &\leq \left\| \sqrt{\text{Id}_{n_{\ell+1}} \otimes \overline{\sigma(z)}} - \sqrt{\text{Id}_{n_{\ell+1}} \otimes K_0^{(\ell+1)}} \right\|^2 \\ &= \left\| \text{Id}_{n_{\ell+1}} \otimes \left(\sqrt{\overline{\sigma(z)}} - \sqrt{K_0^{(\ell+1)}} \right) \right\|^2 \\ &= n_{\ell+1} \left\| \sqrt{\overline{\sigma(z)}} - \sqrt{K_0^{(\ell+1)}} \right\|^2. \end{aligned}$$

Therefore, by (8), we have

$$\mathcal{W}_2^2(h^{(\ell+1)}, H^{(\ell+1)}) \leq n_{\ell+1} \mathbb{E} \left[\left\| \sqrt{\overline{\sigma(G^{\ell})}} - \sqrt{K_0^{(\ell+1)}} \right\|^2 \right].$$

To conclude, we use the following lemma, whose proof is postponed in the next section.

Lemma 4 *Let $X = (X[i])_{i=1}^n$ be independent and identically distributed random variables with values in $\mathbb{R}^k$ and finite fourth moment. Assuming that X is not identically null, let $M = \mathbb{E}[X[1] \otimes X[1]]$ and define the following random variable, taking values in $\mathbb{R}^{k \times k}$,*

$$M_n = \frac{1}{n} \sum_{i=1}^n X[i] \otimes X[i].$$

Then,

$$\mathbb{E} \left[\left\| \sqrt{M_n} - \sqrt{M} \right\|^2 \right] \leq \frac{\mathbb{E} \left[\|X[1] \otimes X[1] - M\|^2 \right]}{n\lambda^+(M)},$$

where $\lambda^+(M) > 0$ denotes the smallest strictly positive eigenvalue of M (which exists since X , hence M , is not identically null). The numerator in the right hand side is finite if X has finite fourth moment.

Indeed, defining $X[i] = \sigma(G^{\ell})[i]$, i.e., $X[i, j] = \sigma(G^{\ell}[i, j])$, the independence assumption is satisfied, because G^{ℓ} is Gaussian centered with covariance $\text{Id}_{n_{\ell} \times n_{\ell}} \otimes K^{(\ell)}[\mathcal{X}]$ hence $G^{\ell}[i_1, j_1]$ and $G^{\ell}[i_2, j_2]$ are not correlated if $i_1 \neq i_2$, which yields independence (which is then preserved after composition with σ). Since σ is not identically null, so is X hence the thesis with $n = n_{\ell}$ yields

$$\mathbb{E} \left[\left\| \sqrt{\overline{\sigma(G^{\ell})}} - \sqrt{K_0^{(\ell+1)}} \right\|^2 \right] \leq \frac{(C^{(\ell+1)})^2}{n_{\ell}},$$

where the finite constant $C^{(\ell+1)}$ is explicitly

$$C^{(\ell+1)} = \sqrt{\mathbb{E} \left[\left\| \sigma(G^{(\ell)})[1] \otimes \sigma(G^{(\ell)})[1] - K_0^{(\ell+1)} \right\|^2 \right] / \lambda + (K_0^{(\ell+1)})}, \quad (19)$$

and depends uniquely on the activation function σ , the input set $\mathcal{X}$ and the weights variances $(c_w^{(i)}, c_b^{(i)})_{i=1}^{\ell}$. The proof of (18) is completed.

Conclusion. We have proved that

$$\mathcal{W}_2(f^{(\ell+1)}, G^{(\ell+1)}) \leq \sqrt{\frac{n_{\ell+1}}{n_{\ell}}} \left(\text{Lip}(\sigma) \sqrt{c^{(\ell)} \mathbb{E} [\|f^{(\ell)} - G^{(\ell)}\|^2]} + C^{(\ell+1)} \right).$$

Recalling that we are considering any probability space where random variables with the same laws as $f^{(\ell)}[\mathcal{X}]$ and $G^{(\ell)}[\mathcal{X}]$ are defined, we consider the infimum over such spaces in the above inequality and conclude that

$$\mathcal{W}_2(f^{(\ell+1)}, G^{(\ell+1)}) \leq \sqrt{\frac{n_{\ell+1}}{n_{\ell}}} \left(\text{Lip}(\sigma) \sqrt{c^{(\ell)}} \mathcal{W}_2(f^{(\ell)}, G^{(\ell)}) + C^{(\ell+1)} \right).$$

Using the inequality provided by the induction assumption we obtain the thesis.

4 Proof of Lemma 4

The argument relies on (3) and the fact that, by the i.i.d. assumption

$$M = \mathbb{E}[M_n], \quad \text{and} \quad \Sigma(M_n) = \Sigma \left(\frac{1}{n} \sum_{i=1}^n X[i] \otimes X[i] \right) = \frac{\Sigma(X[1] \otimes X[1])}{n},$$

so that, by (4),

$$\mathbb{E} [\|M_n - M\|^2] = \mathbb{E} [\text{tr}(\Sigma(M_n))] = \frac{\Sigma(X[1] \otimes X[1])}{n} = \frac{\mathbb{E} [\|X[1] \otimes X[1] - M\|^2]}{n}.$$

However, we need to take into account that M may not to be strictly positive, hence $\lambda(M)$ in (3) could be null. To this aim, we diagonalize $M = UDU^{\tau}$ with U orthogonal, $UU^{\tau} = \text{Id}$ and $D = \sum_{i=1}^k d_i e_i \otimes e_i$ diagonal. Assuming without loss of generality that $d_i > 0$ if $i \leq k'$ and $d_i = 0$ otherwise, for some $k' \leq k$, we consider the random variables $Y[i] = (UX[i])$, $i = 1, \dots, n$ which are i.i.d. with second moment matrix D , since

$$\begin{aligned} \mathbb{E} [Y[1, j_1] Y[1, j_2]] &= \sum_{i_1, i_2=1}^k \mathbb{E} [U[j_1, i_1] X[1, i_1] U[j_2, i_2] X[1, i_2]] \\ &= \sum_{i_1, i_2=1}^k U[j_1, i_1] M[i_1, i_2] U[j_2, i_2] = (UMU^{\tau})[i_1, j_2]. \end{aligned}$$

It follows that $\mathbb{E} [Y[i, j]^2] = d_j = 0$ if $j > k'$, hence $Y[i, j] = 0$ almost surely. Therefore, both D and

$$D_n = UM_nU^\tau = \frac{1}{n} \sum_{i=1}^n U(X[i] \otimes X[i])U^\tau = \frac{1}{n} \sum_{i=1}^n Y[i] \otimes Y[i]$$

have a block structure where only the first $k' \times k'$ entries are not null. It follows that the square root $\sqrt{D_n}$ has the same block structure and by applying (3) only to these blocks we obtain

$$\left\| \sqrt{D} - \sqrt{D_n} \right\| \leq \frac{\|D - D_n\|}{\sqrt{\lambda^+(D)}},$$

where $\lambda^+(D) = \lambda^+(M)$ is strictly positive. □

To conclude, we use that U is orthogonal to obtain that

$$\sqrt{M} = U\sqrt{D}U^\tau \quad \text{and} \quad \sqrt{M_n} = U\sqrt{D_n}U^\tau,$$

as well as the fact that the invariance of the norm $\|UAU^\tau\| = \|A\|$ (for every $A \in \mathbb{R}^{k \times k}$). Thus,

$$\left\| \sqrt{M} - \sqrt{M_n} \right\| = \left\| \sqrt{D} - \sqrt{D_n} \right\| \leq \frac{\|M - M_n\|}{\sqrt{\lambda^+(M)}},$$

squaring both sides and taking expectation the thesis follows.

5 Some remarks on the functional limit

A natural question is to let $k \rightarrow \infty$ in (2), or to consider more generally the neural network (and the Gaussian process) as random variables with values in a (infinite dimensional) space of functions, such as a Hilbert space, e.g., a Lebesgue space $L^2(\mathcal{X}, \mu)$ or even a Banach space, e.g., the space of continuous functions $C^0(\mathcal{X})$, for a compact subset $\mathcal{X} \subseteq \mathbb{R}^{n_0}$ and a finite measure μ on $\mathcal{X}$, e.g., $\mathcal{X} = \mathbb{S}^{n_0-1}$ and μ the surface measure. The main obstacle appears to be Lemma 4, where we use the finite dimensionality assumption on M to obtain a spectral gap, i.e., a bound from below for the smallest strictly positive eigenvalue. Indeed, in infinite dimensions the general bound (3) still holds but we expect that strictly positive eigenvalues of M accumulate towards 0 (M becomes a compact operator). A strategy would be to decompose the action of M into two eigenspaces (corresponding respectively to large and small eigenvalues) and use (3) in the former space and Hölder regularity for the square root in the latter (which gives a worse bound). An alternative strategy would be to rely on the uniform regularity bounds for the process provided by Kolmogorov continuity theorem: arguing as in Bracale et al. (2021), one may prove that, for every compact $\mathcal{X} \subseteq \mathbb{R}^{n_0}$, and $\gamma \in (0, 1)$, letting $C^\gamma(\mathcal{X})$ be the space of γ -Hölder continuous functions (with the usual norm $\|\cdot\|_{C^\gamma}$), then

$$\sup_{(n_l)_{l=1}^{L-1}} \mathbb{E} \left[\left\| f^{(L)} \right\|_{C^\gamma}^2 \right] \leq C(\mathcal{X})$$

for some constant $C(\mathcal{X})$ depending on $\mathcal{X}$ (possibly also γ , n_L and weights and biases variances). A similar bound holds for the Gaussian process $G^{(L)}$. Given any $\varepsilon > 0$, one can find a finite set $\mathcal{K} = \{x_i\}_{i=1}^K$ such that $\inf_{x \in \mathcal{K}} \|x - y\| \leq \varepsilon$ for every $y \in \mathcal{X}$. Then, the triangle inequality gives

$$\|f^{(L)} - G^{(L)}\|_{\mathcal{C}^0} \leq \left(\|f^{(L)}\|_{\mathcal{C}'} + \|G^{(L)}\|_{\mathcal{C}'} \right) \varepsilon^\gamma + \|f^{(L)}[\mathcal{K}] - G^{(L)}[\mathcal{K}]\|.$$

If one could jointly define $f^{(L)}$ and $G^{(L)}$ in such a way that their restrictions on the finite set $\mathcal{K}$ satisfy (2) (with $\mathcal{K}$ instead of $\mathcal{X}$), it would imply an inequality of the type

$$\mathcal{W}_2(f^{(L)}[\mathcal{X}], G^{(L)}[\mathcal{X}]) \leq \inf_{\varepsilon > 0} \left\{ C(\mathcal{X})\varepsilon^\gamma + C(\mathcal{K})\sqrt{n_L} \sum_{\ell=1}^{L-1} \frac{1}{\sqrt{n_\ell}} \right\},$$

where the first constant $C(\mathcal{X})$ in the right hand side is independent of ε , while $C(\mathcal{K})$ depends on ε through $\mathcal{K}$. The Wasserstein distance $\mathcal{W}_2$ above is between probability distributions in $\mathcal{C}^0(\mathcal{X})$ with respect to the cost given by the sup-norm $\|\cdot\|_{\mathcal{C}^0}$. Again, the main issue is that if $\varepsilon \rightarrow 0$ then the set $\mathcal{K}$ becomes large and we expect that the constant $C(\mathcal{K})$ to diverge. It thus appears achievable, although quite technical, to derive uniform bounds on compact sets for the network process and its associated processes.

We finally mention the work (Matsubara et al., 2020), where quantitative bounds have been obtained by considering a non-Gaussian (ridgelet) prior distribution on the network weights, that approximates the Gaussian process in the output space of the network of a shallow Bayesian Neural Network. It is proved there that any Gaussian Process whose covariance regular enough can be approximated by a Bayesian Neural Network.

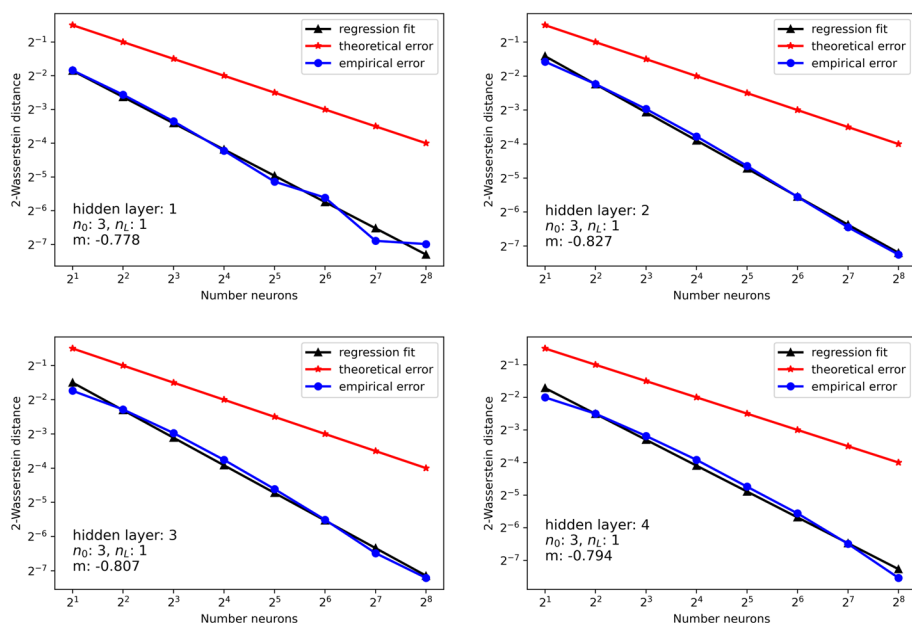


Fig. 2 The above plots refer to the case of a fixed single input, namely $[1, 1, 1] \in \mathbb{R}^3$, and a one-dimensional output $n_L = 1$, while letting the number of hidden layers vary between 1 and 4. The fitted lines (black) even suggest a better rate of convergence. Image produced by the authors as described in Sect. 6

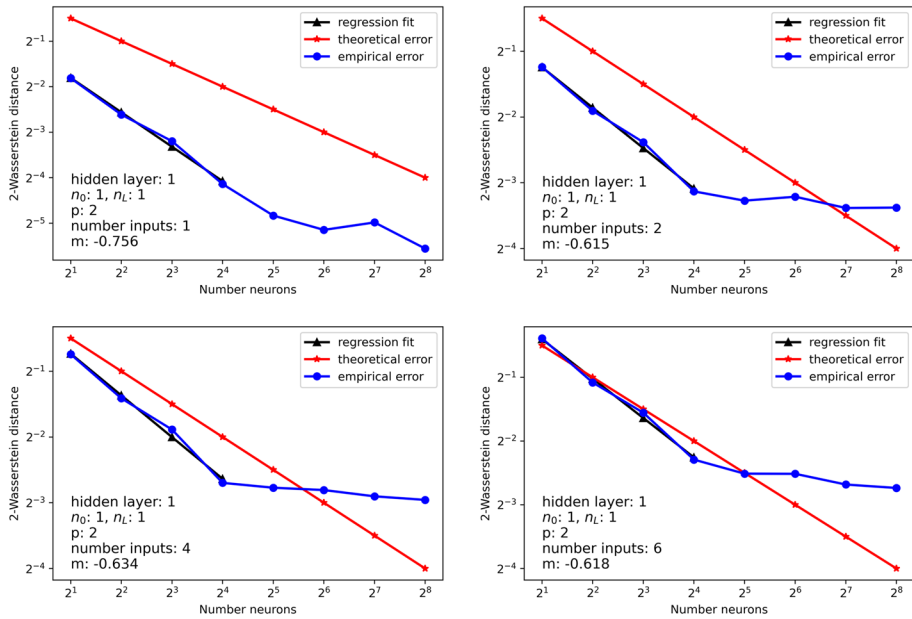


Fig. 3 Keeping a single hidden layer and letting instead the number of inputs vary between 1 and 6 show that the blue lines flatten, because one would need a much larger sample to better approximate the empirical distance with the theoretical one. However, performing a regression on the first half of the graph still suggests convergence rates in agreement with our results. Image produced by the authors as described in Sect. 6

6 Numerical experiments

In this section we present some numerical simulations to investigate the sharpness of Theorem 1. The results are shown in Figs. 2, 3 and 4. Each experiment is structured as follows:

1. we fix the main parameters for a network architecture, i.e. the number of layers L , the input dimension n_0 , output dimension n_L and a set of inputs $\mathcal{X} = \{x_i\}_{i=1,\dots,k} \subseteq \mathbb{R}^{n_0}$,
2. we then set the number of neurons (i.e., the width) in the hidden layers as a parameter $n = n_1 = \dots = n_{L-1}$ (for simplicity we let them be all equal),
3. we sample a large number N of random Gaussian neural networks with weights sampled according to (1) (for simplicity we let the biases be identically zero), and for each sample $j \in \{1, \dots, N\}$ we compute the output $y_j \in \mathbb{R}^{n_L \times k}$ of the network at the inputs $\mathcal{X}$, and use the results to define the empirical distribution

$$\mu_N = \frac{1}{N} \sum_{j=1}^N \delta_{y_j}.$$

4. we sample N i.i.d. points $\{z_j\}_{j=1}^N$ from the centered multivariate Gaussian $\mathcal{N}(K^{(L)}[\mathcal{X}])$, and define a corresponding empirical distribution

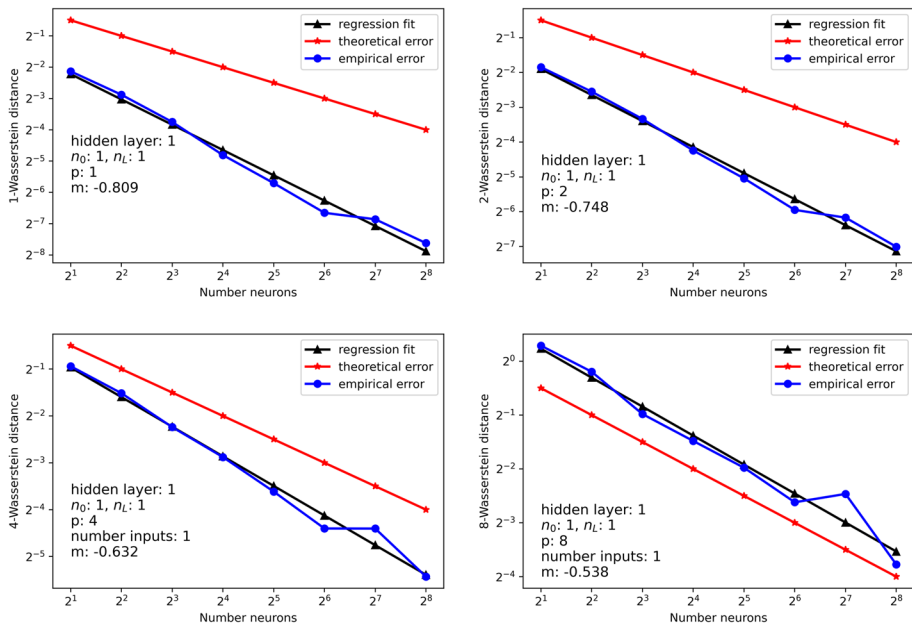


Fig. 4 In the plots above, we fix a single input and single hidden layer structure and compute the Wasserstein distances of order p , for $p \in \{1, 2, 4, 8\}$ (defined by replacing the square in the definition of $\mathcal{W}_2$ with a p -th power and taking the p -th root). The slopes of the regression lines seem to decrease as p grows, getting closer to $-1/2$. The multiplicative constant seems to increase as the black line surpasses the reference red line. Image produced by the authors as described in Sect. 6

$$\nu_N = \frac{1}{N} \sum_{j=1}^N \delta_{z_j},$$

5. we finally compute the Wasserstein distance between the two empirical distributions,

$$\mathcal{W}_2(\mu_N, \nu_N). \quad (20)$$

As N grows, the empirical quantity (20) converges to the corresponding theoretical one, i.e., the left hand side of (2). Precisely, using e.g. the bounds from Fournier and Guillin (2015), we can approximate

$$\left| \mathcal{W}_2(f^{(L)}[\mathcal{X}], G^{(L)}[\mathcal{X}]) - \mathcal{W}_2(\mu_N, \nu_N) \right| \leq CN^{-\alpha},$$

where $\alpha > 0$ depends only on the dimension $n_0 \times k$ and C is a suitable constant (depending on the dimension but also some mild properties of the laws, e.g. their moments). When N is sufficiently large, so that the error term above is much smaller than the right hand side of (2), we expect an inequality of the type

$$\mathcal{W}_2(\mu_N, \nu_N) \leq \frac{C}{\sqrt{n}},$$

for some constant $C > 0$.

As a rule of thumb, we need $N^{-\alpha} \ll n^{-1/2}$. It always holds $\alpha \leq \frac{1}{2}$, as dictated by the CLT, although for higher dimensions the curse of dimensionality appears, making α smaller as the dimension $n_L \times k$ grows. In addition to this, the exact computation of (20) becomes rather time-consuming, and we settled for $N = 10^5$ in our experiments. We used the Python library *POT: Python Optimal Transport* (Flamary et al., 2021) to compute the Wasserstein distance (20) and the *neural-tangents* module (Novak et al., 2020) to compute the kernel $K^{(L)}[\mathcal{X}]$.

In the plots below, the blue lines connect the computed Wasserstein distances between the empirical distributions (in logarithmic scale as a function of the width n of the hidden layers), the black line is a regression line whose slope m is reported at the corner of each plot. The red line is the function $n^{-1/2}$ and it is purely provided for reference. They all show a good agreement with our theoretical result, and may suggest even better rates of convergence, due to a fitted slope m much smaller than $-1/2$.

Author contributions Both authors contributed equally to this work.

Funding A.B. Basteri acknowledges partial support from the French government under management of Agence Nationale de la Recherche as part of the “Investissements d’avenir” program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute), and support from the European Research Council (grant REAL 947908). D.T. acknowledges the MIUR Excellence Department Project awarded to the Department of Mathematics, University of Pisa, CUP I57G22000700001, the HPC Italian National Centre for HPC, Big Data and Quantum Computing - Proposal code CN1 CN00000013, CUP I53C22000690001, the PRIN 2022 Italian grant 2022WHZ5XH - “understanding the LEarning process of QUantum Neural networks (LeQun)”, CUP J53D23003890006, the INdAM-GNAMPa project 2023 “Teoremi Limite per Dinamiche di Discesa Gradiente Stocastica: Convergenza e Generalizzazione”, INdAM-GNAMPa project 2024 “Tecniche analitiche e probabilistiche in informazione quantistica” and the project G24-202 “Variational methods for geometric and optimal matching problems” funded by Università Italo Francese. Research also partly funded by PNRR - M4C2 - Investimento 1.3, Partenariato Esteso PE00000013 - “FAIR - Future Artificial Intelligence Research” - Spoke 1 “Human-centered AI”, funded by the European Commission under the NextGeneration EU programme.

Code availability The code for is available upon request. Its implementation is discussed in Sect. 6.

Declarations

Conflict of interest Not applicable.

Consent for publication The authors of this manuscript consent to its publication.

References

- Agrawal, D., Papamarkou, T., & Hinkle, J. (2020). Wide neural networks with bottlenecks are deep Gaussian processes. *Journal of Machine Learning Research*, 21(175), 1–66.
- Ambrosio, L., Gigli, N., & Savaré, G. (2005). Gradient flows. In *Metric spaces and in the space of probability measures*. Springer.
- Bonis, T. (2020). Stein’s method for normal approximation in Wasserstein distances with application to the multivariate central limit theorem. *Probability Theory and Related Fields*, 178(3), 827–860.
- Borovykh, A. (2019). A Gaussian Process perspective on Convolutional Neural Networks. [arXiv:1810.10798](https://arxiv.org/abs/1810.10798) [cs, stat].
- Bracale, D., Favaro, S., Fortini, S., & Peluchetti, S. (2021). Large-width functional asymptotics for deep gaussian neural networks. In *9th International conference on learning representations, ICLR*
- Cao, W., Wang, X., Ming, Z., & Gao, J. (2018). A review on neural networks with random weights. *Neuro-computing*, 275, 278–287.

- Cho, Y., & Saul, L. (2009). Kernel methods for deep learning. *Advances in Neural Information Processing Systems*, 22.
- Cowan, M.K. (2022). battlesnake/neural. original-date: 2013-08-12T23:46:47Z
- Eldan, R., Mikulincer, D., & Schramm, T. (2021). Non-asymptotic approximations of neural networks by Gaussian processes. In *Conference on learning theory* (pp. 1754–1775). PMLR.
- Eldan, R., Mikulincer, D., & Schramm, T. (2021). *Non-asymptotic approximations of neural networks by Gaussian processes*, 134, 1754–1775.
- Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T. H., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., & Sutherland, D. J. (2021). POT: Python optimal transport. *The Journal of Machine Learning Research*, 22(1), 3571–3578.
- Fournier, N., & Guillin, A. (2015). On the rate of convergence in Wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, 162(3–4), 707–738.
- G. Matthews, A.G., Hron, J., Rowland, M., Turner, R.E., & Ghahramani, Z. (2018). Gaussian process behaviour in wide deep neural networks. In *6th International conference on learning representations, ICLR 2018*, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net,
- Garriaga-Alonso, A., Rasmussen, C.E., & Aitchison, L. (2019). Deep convolutional networks as shallow Gaussian processes
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, vol. 27. Curran Associates, Inc.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- Hanin, B. (2023). Random neural networks in the infinite width limit as Gaussian processes. *The Annals of Applied Probability*, 33(6A), 4798–4819.
- Hemmen, J. L., & Ando, T. (1980). An inequality for trace ideals. *Communications in Mathematical Physics*, 76(2), 143–148.
- Klukowski, A. (2022). Rate of convergence of polynomial networks to gaussian processes. In *Conference on learning theory* (pp. 701–722). PMLR.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- Lee, J., Bahri, Y., Novak, R., Schoenholz, S.S., Pennington, J., & Sohl-Dickstein, J. (2018). Deep neural networks as gaussian processes. In *6th International conference on learning representations, ICLR 2018*, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net,
- Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., & Pennington, J. (2019). Wide neural networks of any depth evolve as linear models under gradient descent. In *Advances in neural information processing systems*, vol. 32. Curran Associates, Inc.
- Matsubara, T., Oates, C. J., & Briol, F. (2020). The ridgelet prior: A covariance function approach to prior specification for Bayesian neural networks. *Journal of Machine Learning Research*, 22, 1–57.
- Mei, S., Montanari, A., & Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33), 7665–7671.
- Narkhede, M. V., Bartakke, P. P., & Sutaone, M. S. (2022). A review on weight initialization strategies for neural networks. *Artificial Intelligence Review*, 55(1), 291–322.
- Neal, R.M. (1996). Priors for infinite networks. In: Neal, R.M. (ed.) *Bayesian Learning for Neural Networks. Lecture Notes in Statistics* (pp. 29–53). Springer.
- Nguyen, P.-M., & Pham, H. T. (2023). A rigorous framework for the mean field limit of multilayer neural networks. *Mathematical Statistics and Learning*, 6(3), 201–357.
- Novak, R., Xiao, L., Hron, J., Lee, J., Alemi, A.A., Sohl-Dickstein, J., & Schoenholz, S.S. (2020). Neural tangents: Fast and easy infinite neural networks in Python. In *International Conference on Learning Representations*
- Peluchetti, S., Favaro, S., & Fortini, S. (2020). Stable behaviour of infinitely wide deep neural networks. In *Proceedings of the Twenty third international conference on artificial intelligence and statistics* (pp. 1137–1146). PMLR. ISSN: 2640-3498.
- Peyré, G., & Cuturi, M. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6), 355–607
- Roberts, D. A., Yaida, S., & Hanin, B. (2022). *The Principles of deep learning theory*. Cambridge University Press.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386.
- Santambrogio, F. (2015). Optimal transport for applied mathematicians. *Birkhäuser, NY*, 55(58–63), 94.

- Sejnowski, T. J. (2018). *The deep learning revolution*. MIT Press.
- Villani, C. (2009). *Optimal Transport: Old and New* vol. 338. Springer.
- Williams, C., & Rasmussen, C.E. (2006). *Gaussian processes for machine learning*, vol. 2, no. 3. MIT Press
- Yang, G. (2019). Wide feedforward or recurrent neural networks of any architecture are Gaussian processes. In *Advances in neural information processing systems 32: Annual conference on neural information processing systems 2019, NeurIPS 2019* (pp. 9947–9960).
- Yang, G. (2019). Wide feedforward or recurrent neural networks of any architecture are gaussian processes. *Advances in Neural Information Processing Systems*, **32**
- Yang, G., & Littwin, E. (2021). Tensor programs iib: Architectural universality of neural tangent kernel training dynamics. In *International conference on machine learning* (pp. 11762–11772). PMLR.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.