Journal of Machine Learning Research 1 (2022) 1-48 Submitted 2/22; Published 3/22

# Neural Networks as Dynamical Systems of Stochastic Fields

**Editor:** -

## Abstract

We develop a framework for over-parametrised neural networks as dynamical systems of stochastic fields. With it, we derive a revised covariance function with squared exponential activations. More importantly, we highlight the first variation covariance function with a given set weight initialisation variances as the determinant of covariance function flow in deep neural networks for wide classes of activation functions. We explain the so-called edge-of-chaos observations and pathological amends in the literature in this framework. Lastly, we derive some conditions on the end-behaviour of activation functions for discrete flow convergence.

**Keywords:** neural tangent kernel, edge of chaos, dynamical systems, stochastic process, deep neural networks

## 1. Introduction

The research for machine learning in neural networks is usually divided into two categories: the typical application research and theoretical research. Application research ranges from speech recognition and synthesis to computer vision, and theoretical research usually seeks to explain why certain network arrangements work. One fundamental kind of arrangement is the deep, fully-connected neural network, which is composed of the more fundamental single hidden layer neural network. It is well-known that in the limit of infinite neurons said network approaches a gaussian process given a prior distribution weights and biases (Rasmussen and Williams (2006)). Cho and Saul (2009) extended that perspective, often categorised as mean field theory, to deep networks with rectified polynomial unit (RePu) activation functions. Using compositions of kernels, however, is simply used, but not proven or justified (de G. Mathews et al. (2018))(Cho and Saul (2009))(Garriga-Alonso et al. (2019)). The study of deep gaussian processes is assumed as a model for the theory of deep neural networks, as Duvenaud et al. (2014) stated and analysed.

In light of this, Poole et al. (2016) and Schoenholz et al. (2017) delivered an analysis of weight initialisation and trainability from the weight perspective, replacing stochastic processes for random variables. We augment that work with covariance functions and explain why their results hold although the analysis does not. It will also be shown how the covariance function composition in Cho and Saul (2009) and Duvenaud et al. (2014) holds with an explicit, non-circular proof, unlike in section C of the appendix of Lee et al. (2018).

Later, we explain their results in the context of deep stochastic processes, simultaneously expanding work in that field by analysing stationary and non-stationary processes by means of the processes' covariance functions and their first variation.

## 2. From Fixed Points to Fixed Functions

We first establish that even a finite-activation neural network with each $w_{ij\ell} \sim \mathcal{N}(0, \sigma_w^2)$ being an independently and identically distributed random variable defines a stochastic process with modest requirements for the activation function. In the limit of infinite activation functions, the network becomes a Gaussian process irrespective of the prior distribution of the weights (Hahn (1977)). Its covariance function is determined by the network inputs, activation function, and the prior distribution over the weights. In order for an activation function, $\alpha$, to generate a valid covariance function, $\alpha \in L_\mu^2(\Omega)$ for the measure $\mu$ with the support $\Omega$.

Each activation output is a sample function of the underlying process for a specified layer upon computation. As such, each layer has its own covariance function defining its corresponding process. Specifically, the layer is a vector of random processes, or a random field, and each layer with the given weight realizations is a sample of the random field. Because the vector entries are independently and identically distributed stochastic processes, their covariance function is repeated along the multivariable covariance matrix diagonal, simplifying the analysis by reducing the analysis from $N$ covariance functions to one.

Given the previous discussion, a single variance or correlation statistic at each layer does not describe the underlying process, especially when describing of the smoothness of the samples. As a result, we study the convergence of covariance functions, not fixed points. We first indicate some inconsistencies regarding fixed point analysis in the machine learning literature.

### 2.1 Fixed Variance and Correlation Inconsistencies

Firstly, it is important to mention that because each entry in the random field is independent, their covariance function will be zero everywhere (the converse is not true). Meaning, that if there were a constant correlation statistic between activation outputs, it would be zero. If one takes the sample view of network outputs, then the variance between them constructs a covariance function. This covariance function can, admittedly, be constant. However, it is so only after several layers, not after the first or second layer, as having a single correlation statistic for all layers implies. Therefore, the only reason the correlation statistic might not be zero over all layers is because of the equivocation between random processes and random variables.

Having established that, assume for the moment that such a statistic is relevant. Continued composition of this correlation map can be analysed as follows: find the critical points of the map and determine whether said points attract or repel points near them. Here we designate fixed points as critical points that attract points near them. Their existence and uniqueness in a local region near the point is guaranteed by the Banach fixed point theorem. One can analyse critical points under function composition using Schröder and Poisson equations. A Taylor approximation is not necessary to analyse function composition dynamics if one makes the following known observations (cite func comp): define the map $F = \{f : x \to f(x)\}$. By definition and with some additional terminology, we put $x_c = f(x_c)$ as a critical point of order one. A critical point of order two would be $x_c = f(f(x_c))$, for example. Taking the derivative of $f$ as $df/dx = f'(x)$ and

of Schröder's equation $\Psi(f(x)) = s\Psi(x))$ as $\Psi'(f(x))f'(x) = s\Psi'(x)$ where $\Psi(x)$ is invertible, and therefore one-to-one in the region near the critical point, we insert the latter as $\Psi'(f(x_c))f'(x_c) = s\Psi'(x_c)$. Because $f(x_c) = x_c$, the $\Psi'$ cancel and $f'(x_c) = s$. Due to the invertibility of $\Psi$, the $\ell^{\text{th}}$ composition of $f$ is $f_\ell(x) = \Psi^{-1}(s^\ell \Psi(x))$. The generally nonlinear function $\Psi$ is found by solving Poisson's equation $\Psi^{-1}(sx) = f(\Psi^{-1}(x))$, which is often an interesting puzzle. Too often, however, the literature dubs the constant $f'(x_c) = s = e^{\log(s)}$ as the Lyapunov exponent claiming that if $s < 1$, $x_c$ is a fixed point −a stable critical point −and if $s > 1$, the map is chaotic −often meaning strongly, or topologically mixing (Katok and Hasselblatt (1995)). When the parameter $s > 1$, it means that points in the neighborhood of $x_c$ are repelled, often to region boundaries or regions of other critical points. Such is the case for the correlation map in Poole et al. (2016), where the critical point at one repels points near it to the region of another that attracts. As a consequence, all $s \leq 1$ for the correlation map, attracting to correlations less than one. Even if the correlation map were a mixing map, that would mean that the correlation of the outputs itself −not the underlying variables it describes −oscillates erratically from one layer to the next, making the correlation of one probable, thereby contradicting the implication that non-unitary correlations define a chaotic region.

This means that random processes −not random variables −constitute the adequate tool to analyse neural networks. Consequently, covariance functions deliver the necessary descriptive heft to acquire meaningful conclusions. We now show how the literature on covariance function composition also equivocates two important concepts in stochastic process theory, namely activation functions and basis functions. Clarifying that leads to a proof of covariance function composition.

## 3. From Covariance Function Composition to Operations on Eigenfunctions and Back

Covariance function composition by insertion of a so-called feature vector pair into the pair of arguments of the covariance function is predicated on two main assumptions: that the feature vector represents the set of basis function of a process and that the resulting kernel is also the covariance function of the process. We will show the two assumptions are indeed what happens in deep neural networks. The embedding of arbitrary functions into covariance functions is pointed out in MacKay (1998), but proving what function to embed and what the resulting covariance function describes remains open. In particular, equating activations with basis functions hampered the proof. We end with an illustrative example using the squared-exponential activation function used in Duvenaud et al. (2014) and replicate their results but prove the attribution of the theory presented here. We also derive another covariance function on the basis of the proof. First, we review some stochastic process theory.

### 3.1 Random Fields

Alluding to Williams (1998), the covariance function for one neuron output process follows

$$k_\alpha(\mathbf{x}_1, \mathbf{x}_2) = \left((2\pi)^{d+1} |\Sigma_w|\right)^{-1/2} \int_\Omega \alpha\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_1\right) \alpha\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_2\right) e^{-\frac{1}{2}\mathbf{w}^{\mathrm{T}}\Sigma_w^{-1}\mathbf{w}}\, d\mathbf{w}\,, \tag{1}$$

where $\mathbf{x}_{1,2} \in \mathbb{R}^{d+1}$ is an augmented deterministic input where $d$ entries are actual inputs and one entry is one to account for the bias weight. The vector $\mathbf{w} \in \mathbb{R}^{d+1}$ has $w_i \sim \mathcal{N}(0, \sigma_w^2), i \in [1, d]$ are i.i.d random variables and $w_{d+1} = b \sim \mathcal{N}(0, \sigma_b^2)$. The matrix $\Sigma_w$ is the covariance matrix with diagonal entries $\sigma_w^2$ up to the $d^{\text{th}}$ entry and the $(d+1)^{\text{th}}$ entry is $\sigma_b^2$. The bars $|\cdot|$ indicate the determinant of the matrix. The function $\alpha(\cdot)$ is the activation function of the layer.

It is known that as $d \to \infty$, the dot product $\mathbf{w}^{\mathrm{T}}\mathbf{x}(t) \to f_t \sim \mathscr{GP}(0, \sigma_w^2 k(t, s))$ approaches a gaussian process with appropriately scaled weights. Also, we parameterise the inputs as $\mathbf{x}_1 = \mathbf{x}(t)$ and $\mathbf{x}_2 = \mathbf{x}(s)$ to emphasise that operations with the input vector entail the same vector at different instances. From this point onward, said gaussian process will be decomposed into its Karhunen-Loève expansion as

$$f_t = \sum_{i=1}^{\infty} \xi_i \phi_i(t) \; . \tag{2}$$

As usual, $\xi_i \sim \mathcal{N}(0, 1)$ and $\phi_i(t)$ are the eigenfunctions, i.e. basis functions, of the process. We shortly prove that $\mathbf{w}^{\mathrm{T}}\mathbf{x}$ is a gaussian process by means of characteristic functions.

$$\begin{aligned}
\varphi_{f_t}(r) &= \mathrm{E}\left[e^{jr\mathbf{w}^{\mathrm{T}}\mathbf{x}(t)}\right] \\
&= (2\pi\,|\Sigma_w|)^{-d/2} \int e^{-\frac{1}{2}\mathbf{w}^{\mathrm{T}}\Sigma_w^{-1}\mathbf{w} + jr\mathbf{w}^{\mathrm{T}}\mathbf{x}(t)}\, d\mathbf{w} \\
&= (2\pi\,|\Sigma_w|)^{-d/2}\, e^{-\frac{1}{2\sigma_w^2}\sum_{i=1}^{d} w_i^2 + jr\sum_{i=1}^{d} w_i x_i(t)}\, d\mathbf{w} \\
&= \prod_{i=1}^{d} \frac{1}{\sqrt{2\pi}\sigma_w} \int e^{-\frac{w_i^2}{2\sigma_w^2} + jr w_i x_i(t)}\, dw_i \\
&= e^{-\frac{\sigma_w^2}{2} r^2 \sum_{i=1}^{d} x_i(t) x_i(s)} \; .
\end{aligned} \tag{3}$$

As one can see by equation (3), any number of deterministic inputs with gaussian weights results in a Gaussian process with covariance function

$$k_{f_t}(t, s) = \sigma_w^2 \sum_{i=1}^{d} x_i(t) x_i(s) \; . \tag{4}$$

If the weights −or inputs −are scaled by $1/\sqrt{d}$ and $d \to \infty$, that would also yield a covariance function. For example, if the inputs in the first layer are numerous and activation functions saturate, then scaling the inputs prevents their premature saturation. Nevertheless, the central limit theorem for stochastic processes is not necessary with deterministic inputs if the prior over the weights is gaussian. Note also that with the characteristic function approach in (3), the embedding theorem for deterministic functions in MacKay (1998) is proven.

For subsequent layers however, because nonlinear activation functions on a gaussian process do not produce, in general, another gaussian process, the layer width must tend to be large, or $d \to \infty$ with $1/\sqrt{d}$ weight scaling in order for the central limit theorem to apply (Hahn (1977)). Replicating the procedure in (3) for a stochastic process would complicate

the procedure to find the covariance function in the sense that the Karhunen-Loève random weights will not be gaussian or that one would have to contend with doubly infinite integrals. In both cases, it is not clear what distribution the Karhunen-Loève random weights should have for a given activation function. As such one may simply invoke the central limit theorem so that $g_t = \lim_{d\to\infty} \mathbf{w}^{\mathrm{T}}\boldsymbol{\alpha}/\sqrt{d}$, where $\boldsymbol{\alpha}$ is the stochastic process $\alpha(\mathbf{w}^{\mathrm{T}}\mathbf{x})$ replicated independently $d$ times. In other words, $\boldsymbol{\alpha}$ is a random field and $g_t \sim \mathscr{GP}(0, \sigma_w^2 k_\alpha(t,s))$.

### 3.2 Activation Functions and Basis Functions

Now is the time to address the equivocation between activation functions and basis functions. As we discussed, $\alpha(\mathbf{w}^{\mathrm{T}}\mathbf{x}(t))$ is a stochastic process whose samples can approximate the covariance function of the underlying stochastic process. For any stationary stochastic process,

$$k_\alpha(t,s) = \lim_{d\to\infty} \frac{1}{d}\sum_{i=1}^{d} \alpha_i(t)\alpha_i(s) \ . \tag{5}$$

With basis functions,

$$k_\alpha(t,s) = \sum_{i=1}^{\infty} \phi_i(t)\phi_i(s) \tag{6}$$

and

$$\lambda_i = \int_T \phi_i^2(t)\ dt \ , \tag{7}$$

where $\lambda_i$ are the eigenvalues, or $L_t^2(T)$ squared norms, of the basis functions. Although equations (5) and (6) seem similar due to their infinite dot product structure, the difference lies in that the scaling for the approximation in equation (5) is not a function of the indices, whereas for equation (6) they are. Furthermore, in order for equation (6) to converge, it is sufficient that

$$\sum_{i=1}^{\infty} \lambda_i < \infty \ , \tag{8}$$

implying that $\lambda_i$ decay. In fact, the speed of decay determines the smoothness of the samples of $\alpha$. These are known facts of operator theory.

Another difference between equations (5) and (6) is that the basis functions $\phi_i$ are not samples of a stochastic process, but completely deterministic. That leads to the topic of function spaces. Since (8) holds, each $\lambda_i < \infty$ and therefore each $\phi_i(t) \in L_t^2(T)$. However, because $\alpha_i(t)$ in equation (5) are samples whose random aspect is the random vector $\mathbf{w}$, it is actually an approximation of the integral (1), and therefore the process $\alpha(t) \in L_\mu^2(\Omega)$, where $\mu$ is the probability measure. Not only are $\phi_i$ and $\alpha_i$ different functions, they are also defined on different function spaces that are distinguished by having different measures defined over different parameter spaces.

A common criterion in random networks to analyse activation functions $\alpha$ is that they be bounded. The convergence of equation (5) motivates said criterion, but since $\alpha_i$ act on an underlying random process dictated by the prior over the weights, one may loosen the boundedness requirement. In fact, $\alpha$ must simply satisfy $\alpha(t) \in L_\mu^2(\Omega)$. From a computational perspective with a gaussian prior over the weights, for example, the requirement also

makes sense: although $\alpha$ might be unbounded away from the mean of $w$, the probability of drawing samples of $w$ with values far away from the mean nears zero. Due to their scarcity compared to the number of samples, $d$, their contribution to the estimate (5) is relatively negligible.

### 3.3 Implications on the Composition of Covariance Functions

Having clarified the difference between activation functions, basis functions, and the equations relating them to the covariance function of a process, we now proceed to propagate that analysis to previous work on the composition of covariance functions, i.e. models for wide and deep networks.

We begin by taking the activation function to be the squared exponential function and have it act on weighted inputs as $\alpha(\mathbf{w}^{\mathrm{T}}\mathbf{x}(t))$. We can construct the covariance function, $k_\alpha$, from equation (1), but we do not know whether the process is gaussian. To assure that, we construct a random field from that one stochastic process, $\boldsymbol{\alpha}(t)$, and perform $\mathbf{w}^{\mathrm{T}}\boldsymbol{\alpha}(t)/\sqrt{d}$ where $\mathbf{w} \in \mathbb{R}^d$ and take $d \to \infty$ to ensure that the process is gaussian with covariance function $\sigma_w^2 k_\alpha$. Said process has a Karhunen-Loève decomposition as in equation (2). We would then be ready to pass that gaussian process through the squared exponential activation function again, ad infinitum, as in deep neural networks.

Since each covariance function has different layer dynamics depending on the inputs, we first consider the case where the inputs are in $\mathbb{R}^2$ and read as $\mathbf{x}(\theta) = [\cos\theta, \sin\theta]^{\mathrm{T}}$. The covariance function of the inputs is therefore

$$k_0(\theta_1, \theta_2) = \sigma_w^2 \cos(\theta_1 - \theta_2) + \sigma_b^2 \ . \tag{9}$$

We define the activation function explicitly as

$$\alpha(x) = \sigma e^{x^2/\ell^2} \ . \tag{10}$$

Using equation (1) with $d = 2$ because of the inputs and setting $\sigma_b = 0$,

$$\begin{aligned}
k_\alpha(\theta_1, \theta_2) &= \left(2\pi\sigma_w^2\right)^{-d/2} \int_{\Omega_w} \alpha\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}(\theta_1)\right) \alpha\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}(\theta_2)\right) e^{-\mathbf{w}^{\mathrm{T}}\mathbf{w}/2\sigma_w^2} \, d\mathbf{w} \\
&= \frac{\sigma^2}{2\pi\sigma_w^2} \int_{\mathbb{R}^2} e^{-\left((w_1\cos\theta_1 + w_2\sin\theta_1)^2 + (w_1\cos\theta_2 + w_2\sin\theta_2)^2\right)/\ell^2 - \left(w_1^2 + w_2^2\right)/2\sigma_w^2} \, d\mathbf{w} \\
&= \frac{\sigma^2}{2\pi\sigma_w^2} \int_{\mathbb{R}^2} e^{-\mathbf{w}^{\mathrm{T}}(I + \Sigma_\theta)\mathbf{w}/2\sigma_w^2} \, d\mathbf{w} \\
&= \sigma^2 \left| I + \Sigma_{\theta_{1,2}} \right|^{-1/2} \\
&= \sigma^2 \left(1 + \frac{4\sigma_w^2}{\ell^2}\left(1 + \frac{\sigma_w^2}{\ell^2}\sin^2(\theta_1 - \theta_2)\right)\right)^{-1/2} \ ,
\end{aligned} \tag{11}$$

where

$$\Sigma_{\theta_{1,2}} = \frac{2\sigma_w^2}{\ell^2} \begin{bmatrix} \cos^2\theta_1 + \cos^2\theta_2 & \frac{1}{2}(\sin 2\theta_1 + \sin 2\theta_2) \\ \frac{1}{2}(\sin 2\theta_1 + \sin 2\theta_2) & \sin^2\theta_1 + \sin^2\theta_2 \end{bmatrix} \ . \tag{12}$$

We put $k_1(\theta_1, \theta_2) = \sigma_w^2 k_\alpha(\theta_1, \theta_2)$ as the covariance function of the stationary gaussian process of the output of the first layer. To find the covariance function of the stochastic

process after applying (10) to the process delineated by $k_1$, we perform the variance operator over all the Karhunen-Loève random variables. Proceeding with the infinite integrals,

$$\begin{aligned} k_2(\theta_1,\theta_2) &= \frac{\sigma_w^2\sigma^2}{\sqrt{2\pi}}\ldots\frac{1}{\sqrt{2\pi}}\int_{\Omega_\xi}\alpha\left(\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{\phi}(\theta_1)\right)\alpha\left(\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{\phi}(\theta_2)\right)e^{-\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{\xi}/2}\,d\boldsymbol{\xi} \\ &= \frac{\sigma_w^2\sigma^2}{\sqrt{2\pi}}\ldots\frac{1}{\sqrt{2\pi}}\int_{\Omega_\xi}e^{-\left(\left(\sum_{i=1}^{\infty}\xi_i\phi_i(\theta_1)\right)^2+\left(\sum_{i=1}^{\infty}\xi_i\phi_i(\theta_2)\right)^2\right)/\ell^2}\,d\mu(\boldsymbol{\xi}) \\ &= \frac{\sigma_w^2\sigma^2}{\sqrt{2\pi}}\ldots\frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}^\infty}e^{-\boldsymbol{\xi}^{\mathrm{T}}(I+\Sigma_\theta)\boldsymbol{\xi}/2}\,d\boldsymbol{\xi} \\ &= \sigma_w^2\sigma^2\left|I+\Sigma_{\theta_{1,2}}\right|^{-1/2} \\ &= \sigma_w^2\sigma^2\left(1+\frac{4}{\ell^2}\left(k_1(0)+\frac{1}{\ell^2}\left(k_1^2(0)-k_1^2(\theta_1,\theta_2)\right)\right)\right)^{-1/2}. \end{aligned} \tag{13}$$

For layers $\nu \geq 1$ and $\sigma_b \neq 0$, covariance function composition applies as

$$k_{\nu+1}\left(\theta_1,\theta_2\right) = \sigma_w^2\sigma^2\left(1+\frac{4}{\ell^2}\left(k_\nu(0)+\frac{1}{\ell^2}\left(k_\nu^2(0)-k_\nu^2(\theta_1,\theta_2)\right)\right)\right)^{-1/2}+\sigma_b^2\,, \tag{14}$$

and the initial condition, so to speak, is $k_1$ as in $k_1(\theta_1,\theta_2) = \sigma_w^2 k_\alpha(\theta_1,\theta_2)+\sigma_b^2$. The accuracy for this stationary covariance function recurrence is explored in Figure 1(b) over various weight variances. For a general non-stationary process, the covariance function is

$$k_{\nu+1}\left(\mathbf{x}_1,\mathbf{x}_2\right) = \sigma_w^2\sigma^2\left(1+\frac{2}{\ell^2}\left(k_{\nu,1,1}+k_{\nu,2,2}+\frac{2}{\ell^2}\left(k_{\nu,1,1}k_{\nu,2,2}-k_{\nu,1,2}^2\right)\right)\right)^{-1/2}+\sigma_b^2\,, \tag{15}$$

where $k_{\nu,i,j}$ means $k_\nu\left(\mathbf{x}_i,\mathbf{x}_j\right)$ and $i,j\in\{1,2\}$.

With (14), one may use function composition theory to analyse the covariance function behaviour in the limit of infinite layers, or $\nu\to\infty$. Specifically, because the process is stationary, we analyse the stationary covariance function along $\theta_1 = -\theta_2 \coloneqq \vartheta$ to acquire $k_0(\vartheta) = \sigma_w^2\cos\left(2\vartheta\right)+\sigma_b^2$. Upon taking the derivative with respect to $k_\nu$,

$$\frac{\partial k_{\nu+1}(\vartheta)}{\partial k_\nu(\vartheta)} = \frac{4\sigma_w^2\sigma^2}{\ell^4}k_\nu(\vartheta)\left(1+\frac{4}{\ell^2}\left(k_\nu(0)+\frac{1}{\ell^2}\left(k_\nu^2(0)-k_\nu^2(\vartheta)\right)\right)\right)^{-3/2}\,, \tag{16}$$

which is the first variation and is valid for non-stationary process covariance functions as well. Whether the composition will converge to a covariance function whose values are the same everywhere in the limit depends on whether the absolute value of equation (16) is less than one for all regions of $\vartheta$. In the case with $\sigma_w = \sigma = \ell = 1$, the fixed function is a finite constant (about 0.56 with a Lyapunov exponent of about $-0.96$), thereby replicating Duvenaud's pathology −also known as the ordered region in weight variance space −albeit with explicit and rigorous formulations for the covariance functions. In fact, we can approximate the equation of the boundary dividing the ordered and stochastic regions with $\sigma = \ell = 1$ for the propagated circle as

$$\sigma_w \approx \frac{1}{2\sqrt{1+\sigma_b^2}}\left(1+4\left(1+\sigma_b^2\right)\right)^{3/4}\,. \tag{17}$$

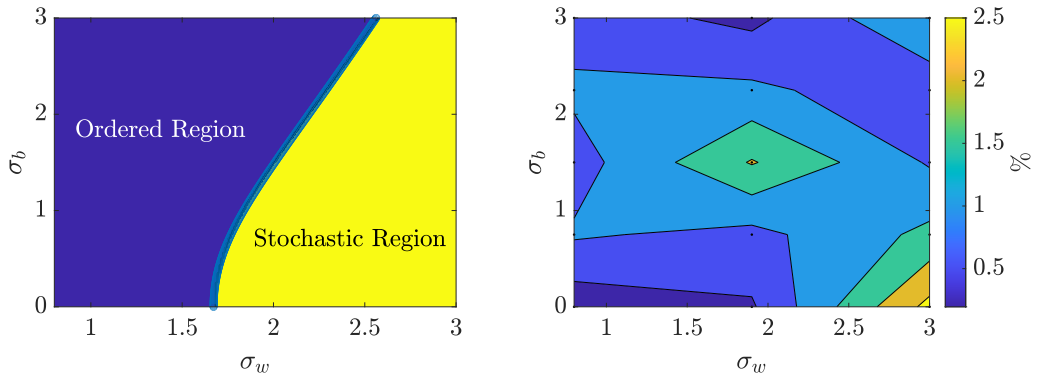



Figure 1: Deep neural network prior with squared exponential activation function described by equations (9), (10), (14), and parameters $\ell = \sigma = 1$. (a) Analytical large layer behaviour, $L = 100$. The criterion for ordered region is $\max(k)/\min(k) - 1 \leq 0.001$. (b) Mean squared error (%) between the numerical and analytical covariance functions. $L = 50$, repeated four times, each layer has $10,000$ neurons. Simulated points are black dots.

The actual boundary involves cubic roots, making it too convoluted for this document.

We further note that, contrary to the much-cited Rasmussen and Williams (2006), the covariance function of an infinitely-wide network with squared exponential activation functions is not a squared exponential, even when properly scaling the weights, as equation (13) shows. It is either a non-stationary covariance function without scaling or zero everywhere with scaled weights with Rasmussen and Williams (2006)'s construction. Luckily, that construction only defines offsets as non-unitary, and the weights are actually all one. Equation (13) is more general, accounting for both non-unitary weights and offsets. This shows that if there is a covariance function formula and the next layer's covariance function is to be found, the arguments passed must be the set of basis functions, i.e. eigenfunctions, of the process. If one passes the activation functions, then one must use equation (5) with the appropriate weight variance operations to approximate $k_\nu$. This example also highlights the difference between activation functions and basis functions: the activations are squared exponentials and the basis functions are Chebyshev-like polynomials.

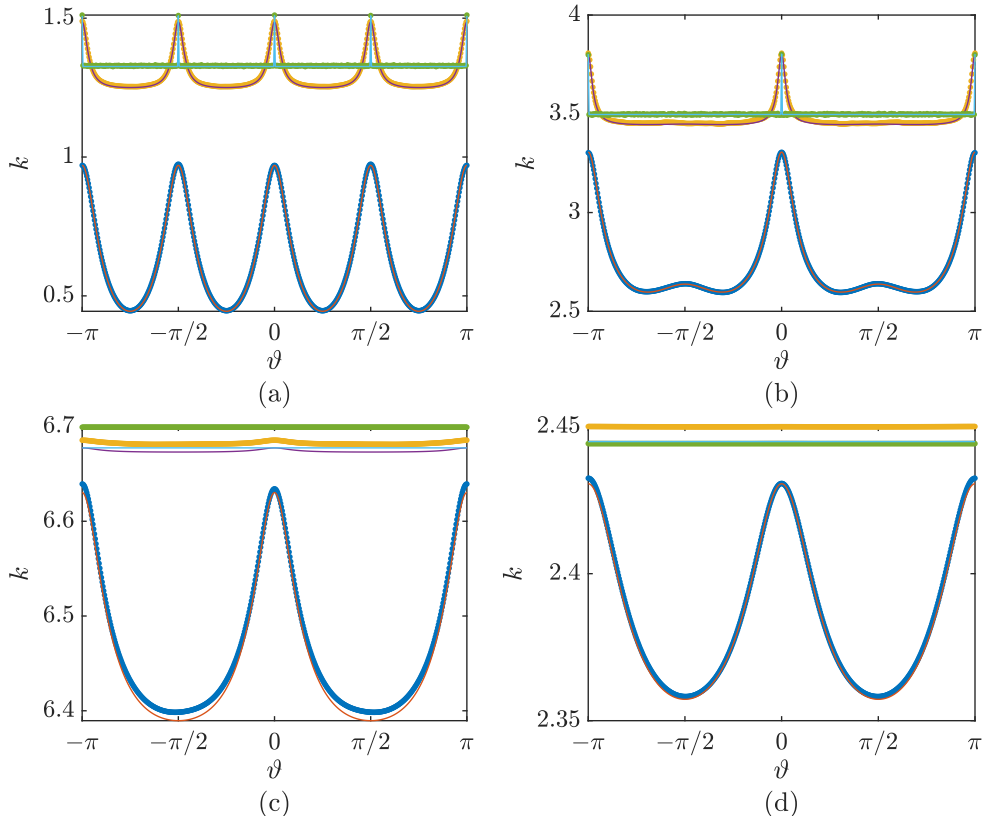



Figure 2: Stationary covariance function layer dynamics for $\alpha(\cdot) = \exp(-(\cdot)^2)$ with the circle over different weight variances. $L = 1$ (blue, red), $L = 5$ (gold, purple), $L = 50$ (green, cyan). Dots are numerical $k$ averaged over 1000 draws and lines are analytical $k$. (a) $\sigma_w = 2, \sigma_b = 0$. (b) $\sigma_w = 2.5, \sigma_b = 1.5$. (c) $\sigma_w = 1.5, \sigma_b = 2.5$. (d) $\sigma_w = 0.8, \sigma_b = 1.5$.

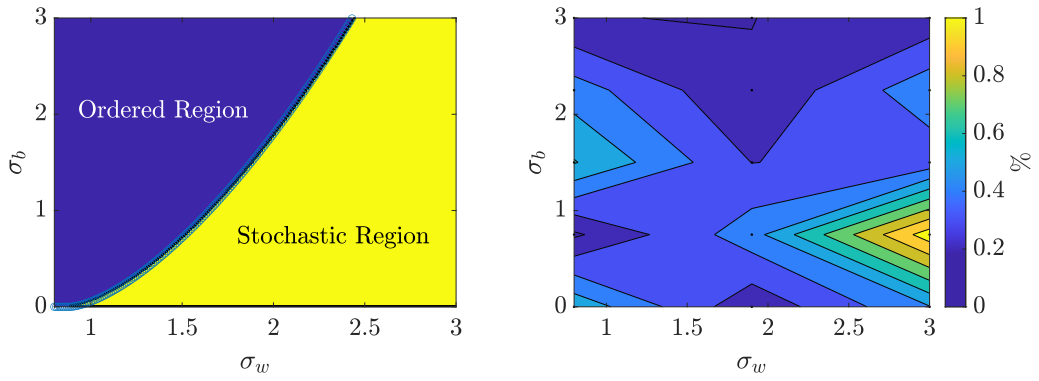
### 3.4 Another Deep Pathology

Although another deep pathology can be shown to occur for the squared exponential activation by simply increasing $\sigma$ and $\sigma_w$ as Figure 1(a) shows, we take the procedure implied in the previous section and apply it to a deep neural net with $\alpha(\cdot) = \text{erf}(\cdot)$ and the $k_0$ in equation (9) with $\sigma_b = 0$. According to Williams (1998), the covariance function is

$$k_\alpha(\theta_1, \theta_2) = \frac{2}{\pi} \arcsin\left(\frac{2\sigma_w^2 \cos(\theta_1 - \theta_2)}{1 + 2\sigma_w^2}\right), \tag{18}$$

and $k_1 = \sigma_w^2 k_\alpha$. The derivation of the covariance function also lends itself to replace the inputs with the basis functions. Recalling equation (7) and that the basis functions are sinusoids in this case, the recursive covariance function is

$$k_{\nu+1}(\theta_1, \theta_2) = \frac{2\sigma_w^2}{\pi} \arcsin\left(\frac{2k_\nu(\theta_1, \theta_2)}{1 + 2k_\nu(0)}\right) + \sigma_b^2 . \tag{19}$$




(a) (b)

Figure 3: Deep neural network prior with error function activation function described by equations (9), (19). (a) Analytical large layer behaviour, $L = 700$. The criterion for ordered region is $\max(k)/\min(k) - 1 \leq 0.001$. (b) Mean squared error (%) between the numerical and analytical covariance functions. $L = 50$, 1000 draws, each layer has 200 neurons. Simulated points are black dots.

Its derivative is

$$\frac{\partial k_{\nu+1}(\vartheta)}{\partial k_\nu(\vartheta)} = \frac{4\sigma_w^2}{\pi\sqrt{(1+2k_\nu(0))^2 - 4k_\infty^2(\nu)}} \,. \tag{20}$$

This covariance function lends itself to a more manageable boundary equation than the squared exponential's covariance function and is shown in equation (21).

$$\sigma_b = \left(\frac{1}{4}\left(\frac{16\sigma_w^4}{\pi^2} - 1\right) - \frac{2\sigma_w^2}{\pi}\arcsin\left(\frac{\frac{16\sigma_w^4}{\pi^2} - 1}{\frac{16\sigma_w^4}{\pi^2} + 1}\right)\right)^{1/2} \tag{21}$$

The covariance function (19), like the squared exponential in equation (14), can also display another kind of pathology hinted in Poole et al. (2016) and Schoenholz et al. (2017). Although there the activation function $\alpha(\cdot) = \tanh(\cdot)$ was used, it is sufficiently close to the $\alpha(\cdot) = \mathrm{erf}(\cdot)$ activation function in $L^2(\mathbb{R}_+)$ that the conclusions drawn for $\mathrm{erf}(\cdot)$ apply to $\tanh(\cdot)$ up to a dilation, $1 + \varepsilon$. Finding the dilation analytically in the chosen function space is easier done numerically, yielding $\varepsilon^* = 0.2028...$, where

$$\varepsilon^* = \min_{\varepsilon > 0} \left\| \tanh\left((1+\varepsilon)x\right) - \mathrm{erf}(x) \right\|_{L^2([0,\infty])} \,. \tag{22}$$

However, because both functions are odd functions with the same end-behaviour, one may approximate equation (22) by finding the $\varepsilon$ that solves

$$\int_0^\infty \tanh\left((1+\varepsilon)x\right) - \mathrm{erf}(x)\; dx = 0 \,. \tag{23}$$

Equation (23) is tractable and $\varepsilon = \sqrt{\pi}\log 2 - 1 \approx 0.2286...$, which has about 13% error from $\varepsilon^*$. The error between the dilations themselves is just about 2%. If one wishes to dilate the

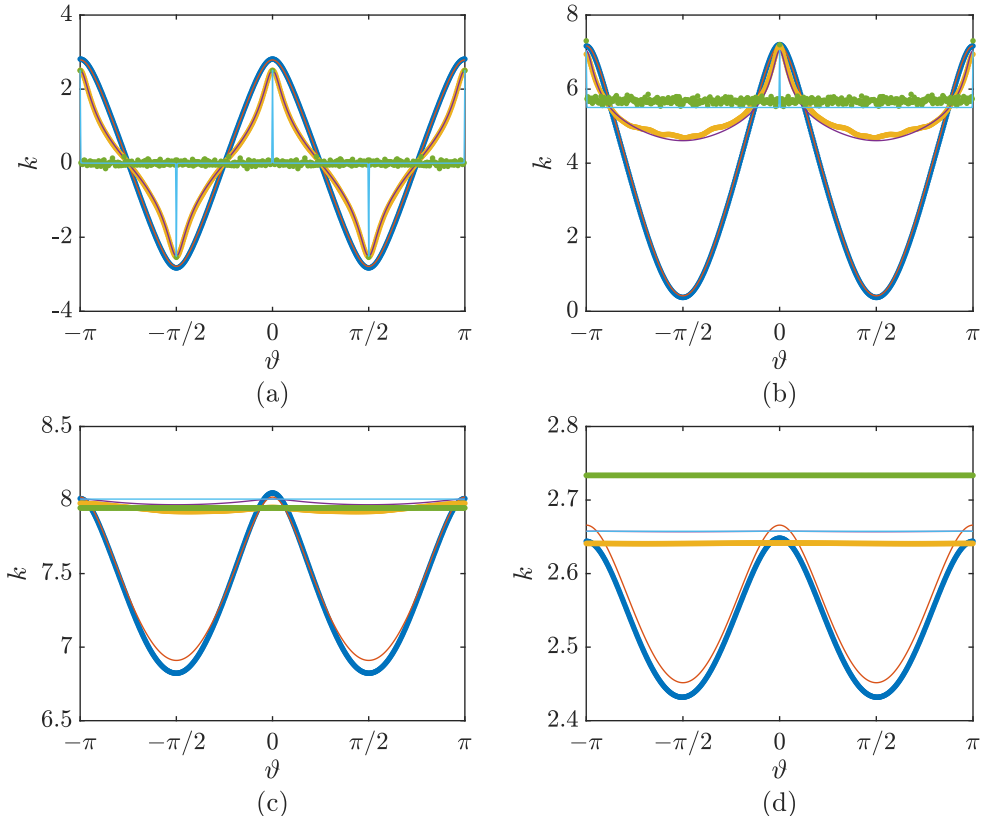



Figure 4: Stationary covariance function layer dynamics for erf(∘) with the circle over different weight variances. $L = 1$ (blue, red), $L = 5$ (gold, purple), $L = 50$ (green, cyan). Dots are numerical $k$ with 11 draws and lines are analytical $k$. (a) $\sigma_w = 2, \sigma_b = 0$. (b) $\sigma_w = 2.5, \sigma_b = 1.5$. (c) $\sigma_w = 1.5, \sigma_b = 2.5$. (d) $\sigma_w = 0.8, \sigma_b = 1.5$.

erf(·) function instead, $\varepsilon \approx -0.186$ and $\varepsilon^* \approx -0.168$, which have a relative error of 11% and the dilations have an error of about 2.2%. The minimum $L^2$ error between the activation functions corresponding to the latter $\varepsilon^*$ is 0.0231.... The covariance function error is about 0.021% with a gaussian prior over the weights. One may also seek a dilation minimising the covariance function error, but that dilation is nearly the same as $1 + \varepsilon^*$ and would not be independent of the prior distribution over the weights. Lastly, any tanh-like function has an $1 + \varepsilon^*$ dilation that minimises the $L^2$ error and nearly minimises the covariance function error because of similar end-behaviour and because said functions can be approximated by lines near the origin. Similar procedures can be constructed for any other known covariance function-activation function pair, reducing the need to estimate the covariance function numerically.

Having established the erf equivalence class, we refer to Figure 4 to signal that covariance functions in the stochastic region contain peaks in the infinite layer limit. These peaks correspond to the process sampling gaussian noise: the other pathology. For limit covariance functions centered at zero, the samples are also centered at zero. For limit covariance

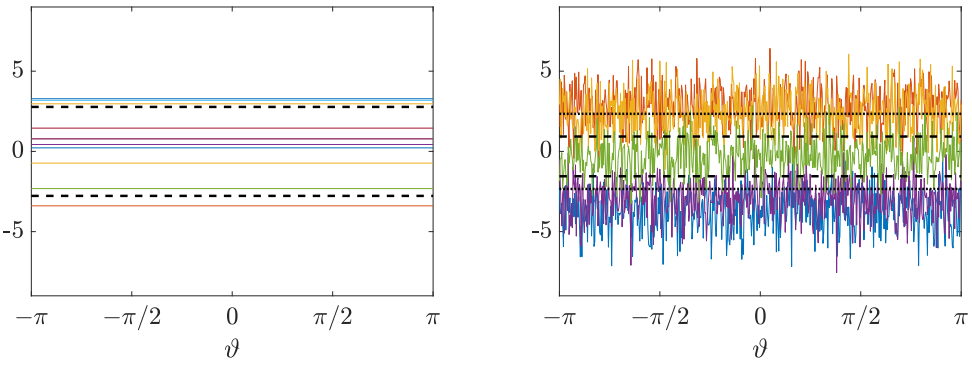



Figure 5: Samples of a deep neural network with erf(·) activations in the (a) ordered region with $\sigma_w = 1.5$, $\sigma_b = 2.5$ and (b) stochastic region with $\sigma_w = 2.5$, $\sigma_b = 1.5$. Dashed lines show the local standard deviation and dotted lines show the standard deviation of the mean of the samples. $L = 50$, 200 neurons.

functions not centered at zero, as in Figure 4 (b), the samples are gaussian noise with a mean whose variance is the covariance function offset. Figure 5 shows sample functions from the arcsin(·) covariance function demonstrating both pathologies.

### 3.5 Analysis of Duvenaud's Amend

Using the procedures from the previous sections we explain why input feedforward amends both pathologies. The recurrent covariance function is

$$k_{\nu+1}(\mathbf{x}_1, \mathbf{x}_2) = \eta\left(k_\nu(\mathbf{x}_1, \mathbf{x}_2)\right) + k_0(\mathbf{x}_1, \mathbf{x}_2) \; . \tag{24}$$

Because $k_0' = k_0$,

$$\frac{\partial k_{\nu+1}}{\partial k_\nu} = \eta'(k_\nu) + k_0 \; , \tag{25}$$

which changes the criterion for the ordered region to read as

$$\eta'(k_\nu) + k_0 < 1 \; . \tag{26}$$

For the stationary squared exponential kernel, $\eta'(k_\nu) = k_{\nu+1}$, whose maximum is one. This reduces the requirement to $k_0 < 0$ for all regions. Since $k_0$ is the dot product of the inputs, we know that when $\mathbf{x}_1 = \mathbf{x}_2$ that dot product will be positive, and the criterion is not met. Therefore the stationary squared exponential kernel with input feed-forward avoids the constant covariance function pathology and the entire weight variance plane is the stochastic region, but with modified smoothness regions for a network implementation. We performed Duvenaud's kernel implementation in equations (24)-(26) so $\sigma_w^2 \equiv 1$ and $k_\alpha = \eta$. We will explain the calculus used in this section in section 5. We lastly note that in this case, because the recurrence reads as $k_{\nu+1} = \exp\left(-1 - \|\mathbf{x}_1 - \mathbf{x}_2\|^2 + k_\nu\right)$, and because

the limiting kernel satisfies $k_{\nu+1} = k_\nu$, solving for $k$ indeed has no closed form, but can be written in terms of the product log function, $\kappa(\cdot)$:

$$k_\infty = -\kappa\left(\exp\left(-1 - \frac{1}{2}\|\mathbf{x}_1 - \mathbf{x}_2\|^2\right)\right) . \tag{27}$$

## 4. Defining Expressivity

Since Adler (1977), and earlier for the one-dimensional parameter and scalar Brownian motion, it is known that certain Gaussian processes have infinite graph lengths over finite intervals. In other words, the dimension of the $\mathbb{R}^{d+1}$ embedding of the Wiener process, the Hausdorff dimension, is greater than $d$ with some dependencies on the parameter dimension (Adler (1977)). For example, the 2-graph of the $d = 1$ Brownian motion with its parameter $t \in \mathbb{R}$ has a dimension of 1.5. Furthermore, Rasmussen and Williams (2006) state that mean square differentiability of the process is determined by the differentiability of the covariance function. Connecting this to an infinite arc-length over a finite interval of a stationary process is straight-forward. Given that a process $f_t = \sum_{i=1}^{\infty} \xi_i \phi_i(t)$,

$$\begin{aligned}
&\mathrm{E}\int_0^1 \sqrt{1 + \left(\frac{df_t}{dt}\right)^2}\, dt \\
&= \mathrm{E}\int_0^1 \sqrt{1 + \sum_{i=1}^{\infty} \xi_i^2 \frac{d\phi_i(t)}{dt}\frac{d\phi_i(s)}{ds} + 2\sum_{i\neq j} \xi_i\xi_j \frac{d\phi_i(t)}{dt}\frac{d\phi_j(s)}{ds}}\, dt \\
&\geq \mathrm{E}\int_0^1 1 + \sum_{i=1}^{\infty} \xi_i^2 \frac{d\phi_i(t)}{dt}\frac{d\phi_i(s)}{ds} + 2\sum_{i\neq j} \xi_i\xi_j \frac{d\phi_i(t)}{dt}\frac{d\phi_j(s)}{ds}\, dt \\
&= 1 + \int_0^1 \frac{\partial^2 k(0)}{\partial t \partial s}\, dt .
\end{aligned} \tag{28}$$

Equation (28) shows that if the covariance function is not differentiable at 0, then the integral explodes, and therefore the expected arc-length of the process $f_t$ explodes as well. From a numerical perspective, the covariance derivative might appear infinite due to the finite resolution of the inputs, but from an analytical perspective it is simply large but finite, which renders the arc-length finite. One may therefore instantly create the illusion of a chaotic process by reducing the length parameter of the covariance function to be less than an input increment. Somewhat conversely, if data resolution remains coarse and the covariance function is not differentiable but has a large distance parameter, the sample functions seem as if they were drawn from a deep network in the ordered region. Furthermore, some processes with non-differentiable covariance functions, like the Ornstein-Uhlenbeck process, have samples that do not appear as if they have infinite arc-length for finite interval when indeed they do. Therefore the appearance of mixing, or chaos, is not an indicator of large, let alone infinite, distances over finite intervals. Since a single-layer neural network has a Gaussian process representation in the limit of infinite activation functions, said processes could have sample functions with infinite graph lengths over finite intervals depending on the covariance function −and therefore depending on the activation function, the prior distribution over the weights, and the inputs. Moreover, any deep network has a covariance

function representation, which can be partitioned into basis functions representing a truncated Gaussian field, which in turn is, by definition, a single-layer network. Considering the previous discussion on the 'compressibility' of deep networks and on the relationship between covariance function length scales and input resolution, it does not aid a practitioner to think about the expressivity of networks. Rather, one should consider which prior adheres best to the data (Kimeldorf and Wahba (1970)); even then, one can maximize the information extracted from the data by changing hyperparameters, e.g. the length scales on covariance functions, after Bayesian updates (Rasmussen and Williams (2006)).

## 5. Covariance Function Flow

According to Rasmussen and Williams (2006) and Kimeldorf and Wahba (1970), the Bayesian update of the prior distribution with the data constitutes 'training' a network with respect to a cost (also objective) function with a quadratic error term and a regression term that penalises high-order derivatives, i.e. complexity in the form of modality. The mean of the posterior is the maximum a priori estimate if the conditional distribution of the data has an exponential convex form (Rasmussen and Williams (2006)). Since gaussian process priors and conditionals have a closed form in terms of covariance functions, one might also approach the gradient explosion and implosion issue by analysing the derivatives of the covariance function over the layers. Duvenaud et al. (2014) hinted at this analysis by exploring the evolution of the tangent space (Jacobian) of each layer, but did not connect it to the gradient explosion and implosion issue explicitly. We now proceed with that analysis here.

### 5.1 Kernel Calculus

One can expect that the covariance function of the process $\alpha'\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}\right)$ will have a relation to the recurrence derivative of equations (20) and (16) as shown in equation (29).

$$\begin{aligned}
\mathrm{E}\left[\nabla_{\mathbf{w}_\nu}\alpha(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,1})\cdot\nabla_{\mathbf{w}_\nu}\alpha(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,2})\right] &= \frac{1}{d_\nu}\mathrm{E}\left[\boldsymbol{\alpha}_{\nu,1}^{\mathrm{T}}\mathrm{E}\left[\alpha'(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,1})\alpha'(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,2})\right]\boldsymbol{\alpha}_{\nu,2}\right] \\
&= \frac{1}{d_\nu}\mathrm{E}\left[\alpha'(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,1})\alpha'(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,2})\right]\mathrm{E}\left[\boldsymbol{\alpha}_{\nu,1}^{\mathrm{T}}\mathbb{I}\boldsymbol{\alpha}_{\nu,2}\right] \\
&= \frac{1}{d_\nu}\mathrm{E}\left[\alpha'_{\nu,1}\alpha'_{\nu,2}\right]\mathrm{E}\left[\left(\sum_{i=1}^{d_\nu}\alpha_{\nu,1,i}\right)\left(\sum_{i=j}^{d_\nu}\alpha_{\nu,2,j}\right)\right] \\
&= \frac{1}{d_\nu}\mathrm{E}\left[\alpha'_{\nu,1}\alpha'_{\nu,2}\right]\mathrm{E}\left[\sum_{i=1}^{d_\nu}\alpha_{\nu,1,i}\alpha_{\nu,2,i}\right] \\
&= \mathrm{E}\left[\alpha'_{\nu,1}\alpha'_{\nu,2}\right]\frac{1}{d_\nu}\sum_{i=1}^{d_\nu}\mathrm{E}\left[\alpha_{\nu,1,i}\alpha_{\nu,2,i}\right] \\
&= \mathrm{E}_{\boldsymbol{\xi}}\left[\alpha'(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,1})\alpha'(\mathbf{w}_\nu^{\mathrm{T}}\boldsymbol{\alpha}_{\nu,2})\right]k_{\alpha,\nu}(\mathbf{x}_1,\mathbf{x}_2),
\end{aligned} \tag{29}$$

where we assumed that $\boldsymbol{\alpha_\nu}$ is a random field process whose entries are independent stochastic processes to allow for recursion. The subscript $\boldsymbol{\xi}$ is written to emphasize that the dot product between the random vector $\mathbf{w}$ and the random field $\boldsymbol{\alpha}_\nu$ is a gaussian process with

a Karhunen-Loève decomposition and that the expectation operator is taken in the sense of equation (13). Next we define $g_\nu$ as

$$g_\nu \coloneqq \sigma_w^2 \mathrm{E}_{\boldsymbol{\xi}} \left[ \alpha'(\mathbf{w}_\nu^{\mathrm{T}} \boldsymbol{\alpha}_{\nu,1}) \alpha'(\mathbf{w}_\nu^{\mathrm{T}} \boldsymbol{\alpha}_{\nu,1}) \right] . \tag{30}$$

Notice that the covariance function of the gradient of the $(\nu+1)^{\text{th}}$ layer in equation (29) is defined in terms of variables in the $\nu^{\text{th}}$ layer. Moreover, the quantity $g_\nu$ modulates the dot product in the $\nu^{\text{th}}$ layer's random field, acting as a metric for a change of coordinates into the tangent space of the $(\nu+1)^{\text{th}}$ field. It consequently functions as a derivative operator on covariance functions between layers. In the literature, equation (29) is referred to as the neural tangent kernel (NTK) (Jacot et al. (2018)) but here the quantity of importance is $g_\nu$, or the first variation covariance function. In previous sections we therefore meant the following:

$$\frac{\partial k_{\nu+1}}{\partial k_\nu} = \sigma_w^2 k'_\alpha(k_\nu) = \eta'(k_\nu) = g_\nu . \tag{31}$$

Scalar dynamical systems theory sheds some light into what happens: we know that if $g_\nu < 1$ for all inputs $\mathbf{x}_1$ and $\mathbf{x}_2$, then the limiting function is reduced to a constant point $c = k_\infty = \Psi^{-1}(0)$, where $\Psi$ is a diffeomorphism over $\mathbf{x}_{1,2}$. This implies that if any $g_\nu > 1$ over $\mathbf{x}_1$ and $\mathbf{x}_2$, $k_\infty \neq c$. The problem in more dimensions becomes choosing what value of $g_\nu$ over the inputs determines the dynamics because $g_\nu : M \times M \to \mathbb{R}$ where $M \subseteq \mathbb{R}^{d_0}$ is the input manifold.

### 5.2 Random Variable Connection

Now we can clarify why the results in Poole et al. (2016) and Schoenholz et al. (2017) could predict Duvenaud's pathology and extend it to weight initialisation: their random variable approach reduces the functions $g_\nu$ and $k_\nu$ to one parameter over all of the input space. The two cases when $\mathbf{x}_1 = \mathbf{x}_2$ and when $\mathbf{x}_1 \neq \mathbf{x}_2$ for $k_\nu$ are taken as the variance and covariance, respectively. Then they normalise the covariance with the variance to construct the correlation parameter. In the limit, the stochastic region $k_\nu$ has two values at those cases, with $k_\nu(\mathbf{x}_1 = \mathbf{x}_2) > k_\nu(\mathbf{x}_1 \neq \mathbf{x}_2)$, giving a correlation number less than one. Since $k_\nu$ is a constant in the ordered region, the variance and covariance are equal and the correlation is one. They found that $g_\nu(\mathbf{x}_1 = \mathbf{x}_2) = 1$, or the correlation dynamics, determines the boundary between the regions and not the variance dynamics because the covariance function determines the layer dynamics of the actual system, as shown here. As such, we expect similar dynamics regarding the gradient explosion and implosion.

### 5.3 Kernel Layer Dynamics

We also do not attribute the conditions on $g_\nu$ as conditions for strong mixing of the underlying samples, but on the layer dynamics of the kernel which in turn describes the underlying process. Now it is the time to answer why the kernel coalesces to roughly a Dirac delta or a constant plane, where scalar analysis reaches its explanatory limits. Scalar composition theory sheds light on what happens in the ordered region, but does not explain why the kernel becomes what it does in the stochastic region. We propose

$$\frac{\partial^2 k_{\nu+1}}{\partial \mathbf{x}_1 \partial \mathbf{x}_2} = \prod_{i=1}^{\nu} g_{\nu+1-i} \tag{32}$$

and therefore along $\mathbf{x}_1 = \mathbf{x}_2$

$$\nabla^2 k_{\nu+1} = \prod_{i=1}^{\nu} g_{\nu+1-i} \ , \tag{33}$$

which does not describe a difference equation in discrete-time (here, the layers) with delay 1 where $\log g_\nu$ determines the dynamics layer after layer. That is due to that $\mathbf{x}_j$ represent the inputs of the network, so that differentiation must take place up to that input layer. If differentiation is done according to the previous random field, then the 'curvature' of the $(\nu+1)^{\text{th}}$ layer is simply $g_\nu$. Nevertheless, if $g_\nu < 1$, $\nabla^2 k \to 0$, which means that $k_\infty$ is flat; if $g_\nu > 1$, $\nabla^2 k \to \infty$, which implies singular curvature at $U = \{(\mathbf{x}_1, \mathbf{x}_2) : \ \mathbf{x}_1 = \mathbf{x}_2\}$. It should be noted that equation (33) holds for any $\mathbf{x}_1$, $\mathbf{x}_2$, so for regions $g_\nu(\mathbf{x}_1 \neq \mathbf{x}_2) < 1$ despite $g_\nu(\mathbf{x}_1 = \mathbf{x}_2) > 1$, $k_\infty$ will be flat where $V = \{(\mathbf{x}_1, \mathbf{x}_2) : \ \mathbf{x}_1 \neq \mathbf{x}_2\}$.

We assumed that the sets $U$ and $V$ remain non-empty throughout the layer dynamics, meaning that parts of the regions that started with $g_\nu < 1$ or $g_\nu > 1$ stay less than one or greater than one throughout the layers, but there is no reason to believe that should be the case. Furthermore, the mean-field approximation merely extended the question as to why $g_\nu(\mathbf{x}_1 = \mathbf{x}_2)$ determines the layer dynamics of the covariance function in the first place. For all of the previous analysis to hold, one has to show that at least subsets of the regions of $g_\nu$ remain in their respective regions of attraction or repulsion throughout the layers. As we will see, the layer dynamics of the kernel itself can also be viewed relative to the input space instead of defining operators over the weight space. From there, the properties we have observed follow.

### 5.3.1 Kernel Flow

We start by defining some components, starting by distinguishing the kernel function from the kernel being fed into it. Let $\eta : \mathbb{R}^2 \to \mathbb{R}$ be the kernel function, or map, and $k : \mathbb{R}^{2d} \to \mathbb{R}$ the kernel, where $\boldsymbol{\omega} = (\boldsymbol{x}, \boldsymbol{y}) \in A$ and $A \subseteq \mathbb{R}^{2d}$. If the inputs are augmented the dimension is simply $2(d+1)$ instead. We also dropped the subscript notation for the inputs and used two different variables for clarity, so $\boldsymbol{x} = \mathbf{x}_1$ and $\boldsymbol{y} = \mathbf{x}_2$. With this construction, $\eta \circ k : \mathbb{R}^{2d} \to \mathbb{R}$ because $\eta = \eta(k(\boldsymbol{\omega}), k(\boldsymbol{x}, \boldsymbol{x}))$ formulaically. It turns out that in general, one may consider $\eta : \mathbb{R} \to \mathbb{R}$ so $\eta = \eta(k(\boldsymbol{\omega}))$ only since $k(\boldsymbol{x}, \boldsymbol{x}) \in k(\boldsymbol{\omega})$, even when taking its first variation.

From general gradient flow we know that critical points, or proto-local extrema, of $\eta$ are fixed points, collected as $\text{Fix}(\eta \circ k) = \{\boldsymbol{\omega}_{*,i}\}$ (Katok and Hasselblatt (1995)). To see this, one can represent the flow of points parametrised with $t$ as $d\boldsymbol{\omega}/dt = G^{-1}\nabla\eta$, but since the metric of $A$ is $G = I$, the equation becomes $d\boldsymbol{\omega}/dt = \partial\eta/\partial k \nabla k$. The gradient flow for each layer has equilibria for points which satisfy $\nabla k_{\boldsymbol{p}} = \mathbf{0}$. Extending the analysis to the $\nu^{\text{th}}$ layer, and noticing that $\partial\eta/\partial k = g$,

$$\frac{d\boldsymbol{\omega}}{dt} = \prod_{i=1}^{\nu} g_{\nu-i+1}(\boldsymbol{\omega}) \nabla_{\boldsymbol{\omega}} k(\boldsymbol{\omega}) \ , \tag{34}$$

suggesting that the critical points of $k$ determine the fixed points of the gradient flow per layer. This does not mean, however, that $k_\nu(\boldsymbol{\omega}_*)$ for $\nabla k_\nu(\boldsymbol{\omega}_*) = \mathbf{0}$ does not change under the iterative map; it just means that those critical points stay critical points throughout the

iterations, and that the values $k_\nu(\boldsymbol{\omega}_*)$ that do remain static or are eventually static over the layers happen on one or more of the critical points. This is an alternative but equivalent approach to Katok and Hasselblatt (1995)'s method of transverse tangent spaces.

We still have not shown the persistence of the values of $g_\nu$ in subsets of $U = \{\boldsymbol{\omega} : \boldsymbol{x} = \boldsymbol{y}\}$ and $V = \{\boldsymbol{\omega} : \boldsymbol{x} \neq \boldsymbol{y}\}$ throughout the layers. Since $g_\nu$ is a per-layer operator, it is bounded loosely as

$$\max_{\boldsymbol{\omega} \in A} g_\nu(\boldsymbol{\omega}) \leq \sigma_w^2 \max_{\zeta \in \Omega} \alpha'(\zeta)^2 \tag{35}$$

per layer. Likewise, $\max k_\nu \leq \sigma_w^2 \max_{\zeta \in \Omega} \alpha(\zeta)^2$, so that given enough samples of the random field, $\max g_\nu$ is attained somewhere in $A$ per layer. Because it is a covariance function, said maximum always happens somewhere or everywhere along the diagonal $\boldsymbol{x} = \boldsymbol{y}$ for each layer. That shows that for every layer, because the activation functions $\alpha$ are the same and given a large enough random field, $\max g_\nu$ is the same for every $\nu > \nu_c$ and happens near the same location in the diagonal. This means that $U$ is non-empty throughout the layers. It is sufficient to analyse $U$ and not $V$ because the maximum of $g_\nu$ determines the inter-layer dynamics in terms of dividing the ordered and stochastic regions. Figures (6) and (7) show the operator $g_\nu$ for different initialisation regions.

Now consider the case for which $\hat{g}_\nu \coloneqq \max g_\nu$ is a constant $1 < c < \infty$ for every $\boldsymbol{\omega} \in A_1$ and $k_\nu$ has quadratically unbounded end-behaviour on $A_1 \subset A$ and therefore no critical points that also function as a global maximum in $A_1$. Then every point in $A_1$ does not have a limit under iterations. A covariance function with such linear end-behaviour in $A$ is ReLu. Its stochastic region never converges to a deep kernel and is flat in $A$ in its ordered region. Say conversely, that $\lim_{\zeta \to \infty} \alpha'(\zeta) = 0$, then because inequality (35) also holds for non-maximal points, $\lim_{|\boldsymbol{\omega}| \to \infty} |g_\nu| = 0$, meaning, by the intermediate value theorem, that there must be a critical point of $g_\nu$ in some $B \subset A$ that is a maximum of $|g_\nu|$ if $g_\nu$ is not zero and the iterations converge to some $k_\infty$.

For convergence to a kernel in its stochastic region, therefore, $\alpha'(\cdot) < \mathcal{O}(1)$ in general. The activation function end-behaviour determines convergence to a deep kernel, $k_\infty$. This can be seen by analysing $\log g$ from $\alpha(\zeta) \sim \mathcal{O}(\zeta^a)$ so that $\alpha'(\zeta) \sim \mathcal{O}(a\zeta^{a-1})$ whence $\log g \sim \mathcal{O}(2 \log a + 2(a-1) \log \zeta) < 0$ for $\zeta \to \infty$. The requirement that $\log g < 0$ comes from the exponential map for iterations. The result is $a < 1$ for convergence in $A$, which means that $\alpha(\cdot)$ need not be bounded, just that $\lim_{\zeta \to \infty} \alpha'(\zeta) = 0$. This is significant because one may construct non-local activation functions resulting in non-local kernels whose convergence to a $k_\infty$ is guaranteed based on the end-behaviour of $\alpha(\cdot)$.

The continuous kernel flow can be modelled as follows. Recall that $k_{\nu+1} = \eta(k_\nu) \approx \eta(k_{\nu-1}) + \eta'(k_{\nu-1})(k_\nu - k_{\nu-1})$. Setting $\varphi_\nu \coloneqq k_\nu - k_{\nu-1}$ and subtracting $k_\nu$ from the first approximation of $k_{\nu+1} = \eta(k_\nu)$ results in

$$\begin{aligned} \varphi_{\nu+1} &\approx g_{\nu-1} \varphi_\nu \\ \log\left(\frac{\varphi_{\nu+1}}{\varphi_\nu}\right) &\approx \log(g_{\nu-1}) \\ \int \frac{d\varphi}{\varphi} &\approx \int_\nu^{\nu+1} \beta(\varphi(t)) dt \\ \dot{\varphi} &\approx \beta(\varphi)\varphi \\ \partial_t \varphi &= \beta(\varphi)\varphi \,, \end{aligned} \tag{36}$$

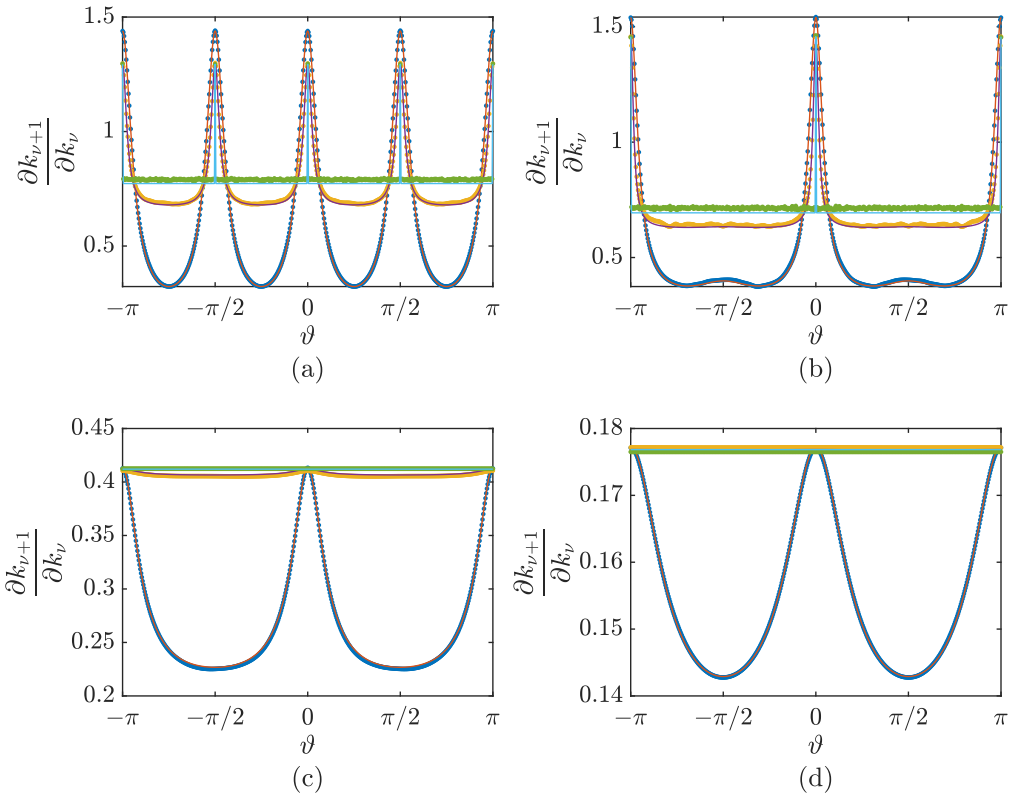



Figure 6: Stationary first variation layer dynamics for $\alpha(\cdot) = \exp(-(\cdot)^2)$ with the circle over different weight variances. $L = 1$ (blue, red), $L = 5$ (gold, purple), $L = 50$ (green, cyan). Dots are numerical $k$ with 1000 draws and lines are analytical $k$. (a) $\sigma_w = 2, \sigma_b = 0$. (b) $\sigma_w = 2.5, \sigma_b = 1.5$. (c) $\sigma_w = 1.5, \sigma_b = 2.5$. (d) $\sigma_w = 0.8, \sigma_b = 1.5$.

where we claim that if we can find a $\beta(\varphi(t))$ satisfying the time integral between layers $\nu$ and $\nu + 1$ that reads $\int \beta(\varphi)dt = \log g_{\nu-1}$, the flow will be determined. Also, because of the latter equation, $\beta(\varphi)$ also determines the flow of $g$. Notice that if $\beta$ is constant we recover the time-one map, which is why $\log g_\nu$ determines the interlayer dynamics up to the critical points of $\eta$. Another candidate flow based on equation (34) and the maximum-preserving property is $\partial_t \eta = |\nabla \eta|^2$ with $\nabla^2 \eta = 0$ except at $\boldsymbol{\omega} = \mathbf{0}$, with maxima converging exponentially according to (36) and the rest of the points as $\mathcal{O}(1/t)$.

As a final comment, for $\eta$-maps whose $k_\infty$ exists, it is known that such systems are stable, meaning that the previous definition of classes of $\eta$-maps based on dilations of known activation functions will converge to kernels near the originals (Katok and Hasselblatt (1995)).

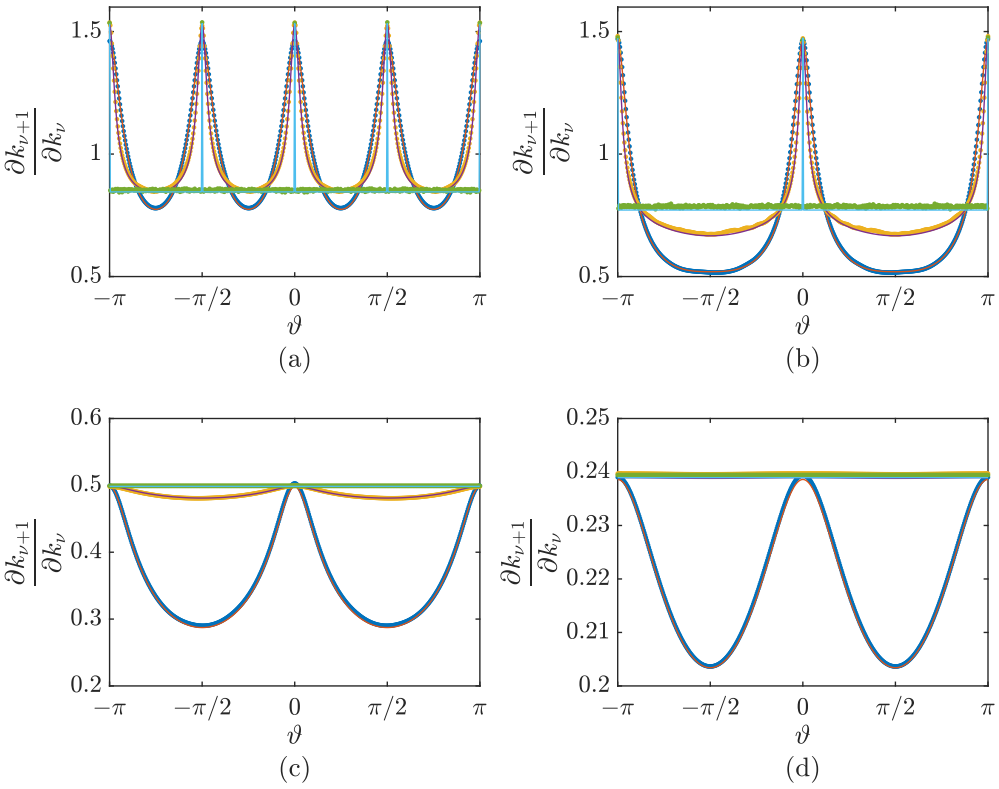



Figure 7: Stationary first variation layer dynamics for $\alpha(\cdot) = \text{erf}(\cdot)$ with the circle over different weight variances. $L = 1$ (blue, red), $L = 5$ (gold, purple), $L = 50$ (green, cyan). Dots are numerical $k$ with 1000 draws and lines are analytical $k$. (a) $\sigma_w = 2, \sigma_b = 0$. (b) $\sigma_w = 2.5, \sigma_b = 1.5$. (c) $\sigma_w = 1.5, \sigma_b = 2.5$. (d) $\sigma_w = 0.8, \sigma_b = 1.5$.

## 6. Conclusion

We have reviewed how the difference between activation and basis functions leads to a proof of covariance function composition if deep neural networks are modelled as dynamical systems of stochastic fields. This, in turn resulted in defining a derivative of covariance functions, whose maximum determines the convergence properties of the covariance function under iterations with itself given a parameter initialisation variance pair. We established that the initial deep network behaviour of a few analytical cases can be extended to other cases where the covariance function of a particular activation function is unknown. This reproduced the results of previous work on random network initialisation within this paper's framework and explained why they occur. We gave some conditions for the convergence of the covariance function and showed that ReLu does not converge. Also, the evolution of the covariance function and its first variation can be explained by kernel flow and is determined by an operation $\beta(\varphi)$. Future work will involve characterising covariance function flow even for thin networks.

## References


Robert J. Adler. Hausdorff Dimension and Gaussian Fields. *The Annals of Probability*, 5 (1):145–151, 1977.

Youngmin Cho and Lawrence K Saul. Kernel Methods for Deep Learning. In *Proceedings of the 22$^{nd}$ International Conference on Neural Information Processing Systems (NIPS)*, pages 342–350, Vancouver, Canada, December 2009. Curran Associates, Inc.

Alexander G. de G. Mathews, Jiri Hron, Mark Rowland, Richard E. Turner, and Zoubin Ghahramani. Gaussian Process Behaviour in Wide Deep Neural Networks. In *Sixth International Conference on Learning Representations*, Vancouver, Canada, April 2018. ICLR.

David Duvenaud, Oren Rippel, Ryan Adams, and Zoubin Ghahramani. Avoiding pathologies in very deep networks. In Samuel Kaski and Jukka Corander, editors, *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pages 202–210, Reykjavik, Iceland, 22–25 Apr 2014. PMLR. URL `https://proceedings.mlr.press/v33/duvenaud14.html`.

Adrià Garriga-Alonso, Carl Edward Rasmussen, and Laurence Aitchison. Deep Convolutional Networks as shallow Gaussian Processes. In *Seventh International Conference on Learning Representations*, New Orleans, LA, May 2019. ICLR.

Marjorie G. Hahn. A Note on the Central Limit Theorem for Square-Integrable Processes. *Proceedings of the American Mathematical Society*, 64(2):331–334, June 1977.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. In *Proceedings of the 32$^{nd}$ Conference on Neural Information Processing Systems (NeurIPS)*, Montréal, Canada, 2018.

Anatole Katok and Boris Hasselblatt. *Introduction to the Modern Theory of Dynamical Systems*, volume 54 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 1995.

George S. Kimeldorf and Grace Wahba. A Correspondence Between Bayesian Estimation on Stochastic Processes and Smoothing by Splines. *The Annals of Mathematical Statistics*, 41(2):459–502, 1970.

Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S. Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep Neural Networks as Gaussian Processes. In *Proceedings of the 6$^{nd}$ International Conference on Learning Representations (ICLR)*, Vancouver, Canada, April-May 2018.

David J.C. MacKay. *Introduction to Gaussian Processes*. May 1998.

Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *30$^{th}$ Conference on Neural Information Processing Systems (NIPS)*, Barcelona, Spain, December 2016. Curran Associates, Inc.

Carl Edward Rasmussen and Christopher K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Samuel S Schoenholz, Justin Gilmer, Surya Ganguli, and Jascha Sohl-Dickstein. Deep Information Propagation. In *International Conference on Learning Representations (ICLR)*, Toulon, France, April 2017.

Christopher K. I. Williams. Computation with Infinite Neural Networks. *Neural Computation*, 10(5):1203–1216, 07 1998. ISSN 0899-7667. doi: 10.1162/089976698300017412. URL `https://doi.org/10.1162/089976698300017412`.