Wide Neural Networks as a Baseline for the Computational No-Coincidence Conjecture

John Dunbar and Scott Aaronson*

UT Austin

Abstract

We establish that randomly initialized neural networks have nearly independent outputs when they have large width, common hyperparameters, and a nonlinear activation function that has zero mean under the Gaussian measure $\mathbb{E}_{z\sim\mathcal{N}(0,1)}[\sigma(z)] = 0$. This includes tanh networks. For our hyperparameters, the zero-mean condition is strict and also proves that ReLU and GeLU networks have well correlated outputs. Because of the nearly independent outputs of tanh and related neural networks, we propose them as a promising construction for the computational no-coincidence conjecture which aims to measure the limits of AI interpretability.

1 Introduction

Significant progress has recently been made in interpreting and understanding the inner workings of AI models, and white-box methods leveraging our understanding have shown potential in applications from detecting backdoors (Lindsey et al., 2025; Goldowsky-Dill et al., 2025) and steering behaviors (Templeton et al., 2024; Turner et al., 2024) to unlearning knowledge (Cloud et al., 2024; Zou et al., 2024). However, model internals have resisted theoretical analysis because the training process is opaque and arbitrary AI models can be cryptographically obfuscated in the worst case (Goldwasser et al., 2024; Christiano et al., 2025). Without a robust understanding, it's difficult to attest to the reliability of our tools and applications. One effort to build our understanding is the computational no-coincidence conjecture from Neyman (2025), a concrete, theoretical conjecture that aims to measure the average case ability of white-box methods.

The conjecture asserts that for random circuits C and a specific rare property $P(C)$ that almost never occurs, there should exist short explanations that prove when the property is true with only a small false positive rate. That is, an explanation should always exist on circuits when $P(C)$ is true, and no explanation should exist on 99% of random circuits. The conjecture uses random reversible circuits $C : \{0,1\}^{3n} \to \{0,1\}^{3n}$ and a property $P(C)$, which we know is rare because of Gay et al. (2025), that inputs ending in n zeros never correspond to outputs ending in n zeros. The explanation and circuit are provided to a polynomial-time verifier that must judge if the explanation is valid and the property is satisfied. If the computational no-coincidence conjecture is true and an accurate verifier exists, it would be evidence that rare behaviors induce detectable structure

*Supported by OpenPhilanthropy

that make interpretability feasible. If the conjecture is false, however, it would be evidence that inscrutable circuits can be common, and we should expect them to exist inside our AI models.

Random reversible circuits are a nice toy model, but our end goal is to understand the no-coincidence conjecture for neural networks. In this work, we make a first step towards that goal by establishing conditions when randomly initialized neural networks behave like random functions and have nearly independent outputs. We propose these random-behaving neural networks as an alternative construction for circuits C, and propose the condition that a set of inputs never correspond to an all-negative output as the rare property $P(C)$.

Our results use the 1st order perturbative expansion and approximation from Roberts et al. (2022) which ignores small-width, $O(1/n^2)$ effects (where n is the network width). Our main result, 4.1, establishes that with appropriate hyperparameters (which are a generalization of Kaiming initialization (He et al., 2015)) and as depth and width scale at a suitable rate, the probability distribution of neural network outputs over the randomness in its weights approaches a standard Gaussian distribution (and thus approaches zero correlation between outputs) if and only if its activation function is nonlinear and has zero mean under the Gaussian measure.

$$\mathbb{E}_{z\sim\mathcal{N}(0,1)}[\sigma(z)] = 0 \tag{1}$$

The zero-mean criteria is satisfied for tanh but not ReLU or GeLU functions, agreeing with prior research that suggests in different ways that tanh networks are biased towards complex functions (Poole et al., 2016) and ReLU networks are biased towards simple functions (Hanin & Rolnick, 2019; Palma et al., 2019; Teney et al., 2024).

2 Problem Setup

Our neural networks are described by the following standard recurrence:

$$\begin{aligned} z_i^{(1)} &= b_i^{(1)} + \sum_{j=1}^{n} W_{ij}^{(1)} x_j \\ \sigma_i^{(\ell)} &= \sigma(z_i^{(\ell)}) \\ z_i^{(\ell)} &= b_i^{(\ell)} + \sum_{j=1}^{n} W_{ij}^{(\ell)} \sigma_j^{(\ell-1)} \text{ for } \ell > 1 \end{aligned} \tag{2}$$

Here, $\sigma : \mathbb{R} \to \mathbb{R}$ is our activation function, n is our network width, and at every layer ℓ, our networks are parameterized by the weight matrix $\{W_{ij}^{(\ell)}\}_{i,j=1,\ldots,n}$ and bias vector $\{b_i^{(\ell)}\}_{i=1,\ldots,n}$. Instead of a dedicated output layer, we will consider the preactivations at an arbitrary layer as the final outputs of our neural networks. Our weights and biases are drawn from a Gaussian distribution with a zero mean and variance set by constants $C_b^{(\ell)}$ and $C_W^{(\ell)}$.

$$\begin{aligned} b_i^{(\ell)} &\sim \mathcal{N}(0, C_b^{(\ell)}) \\ W_{ij}^{(\ell)} &\sim \mathcal{N}(0, C_W^{(\ell)}/n) \end{aligned} \tag{3}$$

A single input to our neural network is a vector $x \in \mathbb{R}^n$, but we'll often need to consider a dataset with many inputs. We'll denote the dataset as $\mathcal{D} = \{x_{i;\alpha}\}_{i=1,\ldots,n;\alpha=1,\ldots}$, where we have used the

notation $x_{i;\alpha} \in \mathbb{R}$ to denote index i of the input vector α. For preactivations, the notation $z^{(\ell)}_{i;\alpha} \in \mathbb{R}$ represents the value before the activation function at neuron index i for input α at layer ℓ.

All expectations will be taken over the random variables $W^{(\ell)}_{ij}$ and $b^{(\ell)}_i$, so to denote general Gaussian expectations with covariance matrix $K \in \mathbb{R}^{m\times m}$, we will use bracket notation:

$$\langle f(z_1, \ldots, z_m)\rangle_K = \int \frac{1}{\sqrt{|2\pi K|}} \exp\left(-\frac{1}{2}\sum_{i,j=1}^{m} (K^{-1})_{ij}\, z_i z_j\right) f(z_1, \ldots, z_m) \prod_{i=1}^{m} dz_i \tag{4}$$

We will drop the subscript for expectations over a standard Gaussian measure $\langle f(z)\rangle = \langle f(z)\rangle_I$. We will also later need the assumption that σ is square integrable under the Gaussian measure.

$$\langle \sigma(z)^2\rangle < \infty \tag{5}$$

3 Near Gaussianity

At the infinite width-limit, it's well known that randomly initialized neural networks become Gaussian processes, and the network's behavior can be completely described by a covariance matrix $K^{(\ell)}$ (Lee et al., 2018). In this setting, demonstrating that a neural network's outputs are independent merely requires proving that the covariance between any two outputs decays to zero. However, neural networks are never infinitely wide in practice. It turns out that at finite width, neural networks are nearly Gaussian, and their non-Gaussian components decay with width (Roberts et al., 2022).

Theorem 3.1 (Roberts et al., 2022)**.** *For a neural network as defined in § 2 with random weights and a fixed set of inputs $\mathcal{D} = \{x_{i;\alpha}\}$ (where $x_{i;\alpha} \in \mathbb{R}$ is index i of datapoint α) and in the nondegenerate case when the covariance matrix is invertible, the preactivations at every layer are distributed as nearly Gaussian:*

$$\begin{aligned} P(z^{(\ell)} \mid \mathcal{D}) \propto \exp\Bigg(&-\frac{1}{2}\sum_{\alpha_1,\alpha_2\in\mathcal{D}} \left(K^{(\ell)} + \frac{1}{n}H^{(\ell)}\right)^{-1}_{\alpha_1\alpha_2} \sum_{i=1}^{n} z^{(\ell)}_{i;\alpha_1} z^{(\ell)}_{i;\alpha_2} \\ &+ \frac{1}{8n}\sum_{\alpha_1,\ldots,\alpha_4\in\mathcal{D}} J^{(\ell)}_{\alpha_1\alpha_2\alpha_3\alpha_4} \sum_{i_1,i_2=1}^{n} z^{(\ell)}_{i_1;\alpha_1} z^{(\ell)}_{i_1;\alpha_2} z^{(\ell)}_{i_2;\alpha_3} z^{(\ell)}_{i_2;\alpha_4} + O\left(\frac{1}{n^2}\right)\Bigg) \end{aligned} \tag{6}$$

Here, the terms $K^{(\ell)}_{\alpha_1\alpha_2}$, $H^{(\ell)}_{\alpha_1\alpha_2}$, and $J^{(\ell)}_{\alpha_1\alpha_2\alpha_3\alpha_4}$ are tensors that depend on the inputs $\mathcal{D}$, layer ℓ, and activation function σ, and the $O(1/n^2)$ term contains additional tensors and instances of z. The covariances $K^{(\ell)}_{\alpha_1\alpha_2}$ of this nearly Gaussian distribution are determined by the following equations:

$$\mathbb{E}[z^{(\ell)}_{i_1;\alpha_1} z^{(\ell)}_{i_2;\alpha_2}] = \delta_{i_1 i_2} K^{(\ell)}_{\alpha_1\alpha_2} + O\left(\frac{1}{n}\right) \tag{7}$$

$$K^{(1)}_{\alpha_1\alpha_2} = C^{(1)}_b + \frac{C^{(1)}_W}{n}\sum_{j=1}^{n} x_{j;\alpha_1} x_{j;\alpha_2} \tag{8}$$

$$K^{(\ell)}_{\alpha_1\alpha_2} = C^{(\ell)}_b + C^{(\ell)}_W \langle \sigma(z_{\alpha_1})\sigma(z_{\alpha_2})\rangle_{K^{(\ell-1)}} \text{ for } \ell > 1 \tag{9}$$

For completeness, we prove a simplified form of Theorem 3.1 in Appendix A. For the full proof, see Roberts et al. (2022). From the theorem we see that correlations are primarily governed by the width-independent, second-order terms $K^{(\ell)}_{\alpha_1\alpha_2}$.

4 Decaying Covariances

//TODO forgot that for K to be invertible, size of dataset must be smaller than size of data Since the Gaussian component of (6) does not decay with width, the independence of our neural network's outputs depends on the Gaussian covariances decaying to zero. We now leave generality to analyze when this decay occurs. Consider the special case where input vectors are normalized to $\sqrt{n}$ and the hyperparameters C_b and C_W are set as follows.

$$\begin{aligned} \frac{1}{n}\sum_{i=1}^{n}(x_{i;\alpha})^2 &= 1 \\ C_W^{(1)} &= 1 \\ C_W^{(\ell)} &= \frac{1}{\langle \sigma(z)^2 \rangle} \text{ for } \ell > 1 \\ C_b^{(\ell)} &= 0 \end{aligned} \tag{10}$$

We'll refer to these hyperparameters in combination with the input normalization as a critical tuning because they prevent the diagonal terms $K_{\alpha\alpha}^{(\ell)}$, the variance of a single input's preactivations, from growing or decaying at every layer.

$$\begin{aligned} K_{\alpha\alpha}^{(1)} &= \frac{1}{n}\sum_{i=1}^{n}(x_{i;\alpha})^2 = 1 \\ K_{\alpha\alpha}^{(\ell)} &= \frac{\langle \sigma(z)^2 \rangle_{K_{\alpha\alpha}^{(\ell-1)}}}{\langle \sigma(z)^2 \rangle} = 1 \text{ for } \ell > 1 \end{aligned} \tag{11}$$

These settings are also not uncommon in practice. The weight variance C_W is a generalization of Kaiming initialization (He et al., 2015) which was also designed with the intent of keeping variances stable.

Since the covariances $K_{\alpha_1\alpha_2}^{(\ell)}$ are a function of two inputs α_1, α_2 and have no dependence on the other inputs, it's sufficient to consider datasets with only two inputs. There, $K^{(\ell)}$ is a 2×2 matrix and it becomes clear that, because the diagonal terms are constant ($K_{\alpha\alpha}^{(\ell)} = 1$), the off-diagonal terms at any layer are a function of their counterparts at the previous layer, $K_{\alpha_1\alpha_2}^{(\ell)} = \mathcal{C}(K_{\alpha_1\alpha_2}^{(\ell-1)})$ for some $\mathcal{C} : \mathbb{R} \to \mathbb{R}$. We will see that as we progress through the layers, the off-diagonal term approaches a fixed point based on the activation function σ.

Theorem 4.1. *In a neural network as defined in § 2 with random weights, critical hyperparameters from* (10)*, nonlinear activation function σ, and a set of fixed inputs $\mathcal{D}$ containing no scalar multiples (i.e. $|K_{\alpha_1\alpha_2}^{(1)}| \neq 1$ for $\alpha_1 \neq \alpha_2$), preactivations are distributed as follows with exponentially decaying covariance if and only the activation function has zero mean under the Gaussian measure, $\langle \sigma(z) \rangle = 0$.*

$$\begin{gathered} P(z^{(\ell)} \mid \mathcal{D}) \propto \exp\left(-\frac{1}{2} \sum_{\alpha_1,\alpha_2 \in \mathcal{D}} (K^{(\ell)})^{-1}_{\alpha_1\alpha_2} \sum_{i=1}^{n} z_{i;\alpha_1}^{(\ell)} z_{i;\alpha_2}^{(\ell)} + O\left(\frac{1}{n}\right) \right) \\ K_{\alpha\alpha}^{(\ell)} = 1 \qquad K_{\alpha_1\alpha_2}^{(\ell)} = \exp(-\Omega(\ell)) \text{ for } \alpha_1 \neq \alpha_2 \end{gathered} \tag{12}$$

Conversely, when the activation function has a nonzero mean, the covariance does not decay to zero, $\lim_{\ell\to\infty} K_{\alpha_1\alpha_2}^{(\ell)} > 0$.

Although the width dependent $O(1/n)$ terms may be a function of ℓ, which we have treated as a constant so far, there exists a suitable way[1]to scale both n and ℓ simultaneously such that the $O(1/n)$ term becomes less than exponentially small in ℓ.

Corollary 4.1. *For a depth ℓ neural network defined the same as in Theorem 4.1 at sufficiently large width n, the probability distribution of outputs $P(z^{(\ell)} \mid \mathcal{D})$ is within $\exp(-\Omega(\ell))$ total variation distance from a standard Gaussian distribution.*

Proof. This corollary follows from rewriting the diminishing components in Theorem 4.1 in terms of ℓ at sufficiently large width scaling. First, using the perturbed matrix inverse identity, we can simplify the expression for $(K^{(\ell)})^{-1}_{\alpha_1\alpha_2}$.

$$K^{-1} = (I + \exp(-\Omega(\ell)))^{-1} = I + \exp(-\Omega(\ell)) \tag{13}$$

Where I is the identity matrix. We then have the following:

$$\begin{aligned} P(z^{(\ell)} \mid \mathcal{D}) &\propto \exp\left(-\frac{1}{2}\sum_{\alpha\in\mathcal{D}}\sum_{i=1}^{n}(z^{(\ell)}_{i;\alpha})^2 + \exp(-\Omega(\ell))\right) \\ &= \frac{\exp\left(-\frac{1}{2}\sum_{\alpha\in\mathcal{D}}\sum_{i=1}^{n}(z^{(\ell)}_{i;\alpha})^2 + \exp(-\Omega(\ell))\right)}{\int \exp\left(-\frac{1}{2}\sum_{\alpha\in\mathcal{D}}\sum_{i=1}^{n}(z_{i;\alpha})^2 + \exp(-\Omega(\ell))\right)\prod_{\alpha,i} dz_{i;\alpha}} \\ &= \frac{1}{\sqrt{(2\pi)^n}}\exp\left(-\frac{1}{2}\sum_{\alpha\in\mathcal{D}}\sum_{i=1}^{n}(z^{(\ell)}_{i;\alpha})^2\right)(1+\exp(-\Omega(\ell))) \end{aligned} \tag{14}$$

In the last equality, we Taylor expanded around $\exp(-\Omega(\ell)) = 0$. In this form, the total variation distance from a standard Gaussian is clear.

$$\int\left(P(z^{(\ell)} \mid \mathcal{D}) - \frac{1}{\sqrt{(2\pi)^n}}\exp\left(-\frac{1}{2}\sum_{\alpha\in\mathcal{D}}\sum_{i=1}^{n}(z^{(\ell)}_{i;\alpha})^2\right)\right)\prod_{\alpha,i} dz_{i;\alpha} = \exp(-\Omega(\ell)) \tag{15}$$

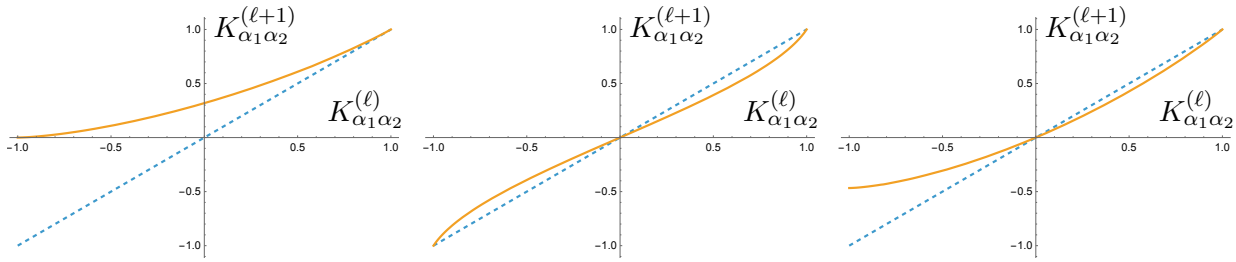
□

Before proving Theorem 4.1, let's build up a few useful properties of the covariances $K^{(\ell)}_{\alpha_1\alpha_2}$. Let $\mathcal{C} : \mathbb{R} \to \mathbb{R}$ be the map from the covariance at one layer to the covariance at the next.

$$\begin{aligned} K^{(\ell+1)}_{\alpha_1\alpha_2} &= \mathcal{C}(K^{(\ell)}_{\alpha_1\alpha_2}) \\ \mathcal{C}(K^{(\ell)}_{\alpha_1\alpha_2}) &= C_b + C_W\langle\sigma(z_1)\sigma(z_2)\rangle_\Sigma = \frac{\langle\sigma(z_1)\sigma(z_2)\rangle_\Sigma}{\langle\sigma(z)^2\rangle} \\ \Sigma &= \begin{pmatrix} 1 & K^{(\ell)}_{\alpha_1\alpha_2} \\ K^{(\ell)}_{\alpha_1\alpha_2} & 1 \end{pmatrix} \end{aligned} \tag{16}$$

In Figure 1 we plot $\mathcal{C}$ for the ReLU and tanh activation functions. The covariance between preactivations after many layers is equal to the repeated application of $\mathcal{C}$ on the initial covariance $K^{(1)}_{\alpha_1\alpha_2}$, and we can see in the plots that for ReLU and tanh, the repeated application of $\mathcal{C}$ appears to bring most initial covariances to a fixed point at 1 or 0. In § 4.2 we will prove that this is not a coincidence and that all nonlinear activation functions bring covariances to a fixed point. Eventually, to prove Theorem 4.1 we will need to understand exactly when the fixed point is at 0.

[1]We conjecture that the necessary scaling is most likely exponential, $n = \exp(O(\ell))$, because the next-order tensors H and V from Theorem 3.1 grow at most exponentially in ℓ at the critical tuning (10).



$\sigma(x) = \mathrm{ReLU}(x)$ $\sigma(x) = \tanh(4x)$ $\sigma(x) = \mathrm{ReLU}(x) - \frac{1}{\sqrt{2\pi}}$

Figure 1: Graphs of $K^{(\ell+1)}_{\alpha_1\alpha_2} = \mathcal{C}(K^{(\ell)}_{\alpha_1\alpha_2})$ for when the activation function is ReLU (left), $\tanh(4x)$ (center), or a shifted ReLU (right). The $y = x$ line in dotted blue is included for comparison. For ReLU, the repeated application of $\mathcal{C}$ brings initial points towards the stable fixed point at $K^{(\ell)}_{\alpha_1\alpha_2} = 1$. This means the preactivations on different inputs become more and more correlated as depth increases. For tanh (for which the $4x$ scaling was chosen to emphasize the effect and doesn't change it qualitatively), the repeated application of $\mathcal{C}$ brings initial points towards 0 unless they start at 1 or -1. This means preactivations on different inputs usually become less and less correlated as depth increases, but they stay identical if they started identical up to a sign. A similar effect occurs for a ReLU shifted to have zero mean under the Gaussian measure

4.1 Hermite Decompositions

To understand (16) and the behavior of $\mathcal{C}$, we can decompose our activation function σ into Hermite polynomials He_n which are orthogonal under Gaussian weighting (Bateman & Erdélyi, 1953).

$$\begin{aligned} \mathrm{He}_0(x) &= 1 \\ \mathrm{He}_1(x) &= x \\ \mathrm{He}_2(x) &= x^2 - 1 \\ \mathrm{He}_3(x) &= x^3 - 3x \\ \langle \mathrm{He}_n(z)\mathrm{He}_m(z)\rangle &= \delta_{nm} n! \\ \|\mathrm{He}_n\|^2 = \langle \mathrm{He}_n(z)^2\rangle &= n! \end{aligned} \tag{17}$$

Any function that is square integrable under the Gaussian measure ($\langle \sigma(z)^2\rangle < \infty$) can be decomposed into Hermite polynomials. This is a weak condition that, as mentioned before, we assume to be true of our activation functions.

$$\sigma(z) = \sum_{n=0}^{\infty} a_n \mathrm{He}_n(z) \tag{18}$$

$$a_n = \frac{1}{\|\mathrm{He}_n\|^2} \langle \sigma(z)\mathrm{He}_n(z)\rangle \tag{19}$$

For correlated Gaussians, we can make a similar statement. The following identity derives from Mehler's equation. See Appendix B for the derivation.

$$\begin{aligned} \langle \mathrm{He}_n(z_1)\mathrm{He}_m(z_2)\rangle_\Sigma &= \delta_{nm}\, n!\, k^n \\ \Sigma &= \begin{pmatrix} 1 & k \\ k & 1 \end{pmatrix} \end{aligned} \tag{20}$$

After decomposing our activation function into Hermite polynomials, our map $\mathcal{C}$ greatly simplifies into a polynomial.

$$\begin{aligned} \mathcal{C}(k) &= \frac{\langle \sigma(z_1)\sigma(z_2)\rangle_\Sigma}{\langle \sigma(z)^2\rangle} \\ &= \frac{1}{\langle \sigma(z)^2\rangle} \left\langle \left(\sum_{n=0}^{\infty} a_n \mathrm{He}_n(z_1)\right)\left(\sum_{n=0}^{\infty} a_n \mathrm{He}_n(z_2)\right)\right\rangle_\Sigma \\ &= \frac{1}{\langle \sigma(z)^2\rangle} \sum_{n=0}^{\infty} (a_n)^2\, n!\, k^n \end{aligned} \tag{21}$$

Using this polynomial, we will prove that all nonlinear activation functions have an attractive fixed point.

4.2 Fixed Points

In this section we'll build up to and prove a lemma (Lemma 4.3) which states that the repeated application of $\mathcal{C}$ always reaches a single fixed point for any initial covariance $K^{(1)}_{\alpha_1\alpha_2} \in (-1, 1)$. Once that is established, Theorem 4.1 immediately follows because our fixed point is at 0 exactly when $\mathcal{C}(0) = 0$, and that is equivalent to when the first coefficient $a_0 = \langle \sigma(z)\rangle$ in (21) is 0. Although some lemmas require a nonaffine instead of nonlinear activation function, the affine, nonlinear case is simple to handle separately.

Lemma 4.1. *The function $\mathcal{C}$ is convex and non-negative in the $[0, 1]$ domain, and strictly convex in that domain when the activation function σ is nonaffine.*

Proof. The convexity follows because $\mathcal{C}$ in (21) is a positive combination of convex monomials. For the strict convexity, note that $\mathrm{He}_0(x) = 1$ and $\mathrm{He}_1(x) = x$, so Hermite decompositions with only the first two terms represent exactly the affine functions. Thus, the decomposition of any nonaffine function has some non-zero coefficient in $\{a_n : n > 1\}$ which multiplies a strictly convex monomial $\{k^n : n > 1\}$ in (21), causing $\mathcal{C}$ to be strictly convex. □

Lemma 4.2. *The function $\mathcal{C}$ is above the diagonal, $\mathcal{C}(k) > k$, in the $(-1, 0)$ domain when the activation function σ is nonlinear.*

Proof. First, we trivially have $\mathcal{C}(1) = 1$ because neural networks are deterministic, and identical activations stay identical. At the point $(1, 1)$, our monomials in (21) are all 1 and we see that the coefficients $(a_n)^2 n!/\langle \sigma(z)^2\rangle$ are actually a convex combination. Thus in the $(-1, 0)$ domain, $\mathcal{C}$ is above the diagonal because it is a convex combination of monomials that are either at the diagonal (for the linear term) or above it. A monomial that's above the diagonal is always included in the combination when the activation function is nonlinear. □

Lemma 4.3. *For any starting input $k_0 \in (-1, 1)$, the repeated application of $\mathcal{C}$ on k_0 approaches a unique fixed point $k^* \in [0, 1]$ when the activation function σ is nonaffine.*

Proof. Let $\mathcal{C}' = \frac{d}{dk}\mathcal{C}(k)$ be the derivative of $\mathcal{C}$. Let's split the proof into cases based on the value of the derivative at $k = 1$.

Case 1: If $\mathcal{C}'(1) \leq 1$, then $\mathcal{C}$ is always above the diagonal (except at $\mathcal{C}(1) = 1$). This is because Lemma 4.2 directly applies in the $(-1, 0)$ domain and because of strict convexity in the $[0, 1)$ domain. Thus, the repeated application of $\mathcal{C}$ increases any initial value k_0 up to the fixed point at the domain's boundary, $\mathcal{C}(1) = 1$.

$$\mathcal{C}(k) > \mathcal{C}'(1)(k - 1) + 1 \geq k \tag{22}$$

Case 2: If $\mathcal{C}'(1) > 1$, conversely, then there's a second fixed point k^* in the domain $[0, 1)$. This is because (21) causes $\mathcal{C}(0)$ to be at or above the diagonal, and the steep derivative $\mathcal{C}'(1) > 1$ causes $\mathcal{C}(1 - \varepsilon)$ for small ε to be below the diagonal. Above at one point and below at another, $\mathcal{C}$ must be exactly on the diagonal somewhere in between. That fixed point is also unique because along with $\mathcal{C}(1) = 1$, a third fixed point would violate the strict convexity in the $[0, 1)$ domain.

Now that we've proven the existence of a fixed point $k^* \in [0, 1)$, we must prove the convergence to it. In the $(-1, 0)$ domain, Lemma 4.2 directly applies and the repeated application of $\mathcal{C}$ increases values until they reach the $[0, 1)$ domain. In the part of the $[0, 1)$ domain where $\mathcal{C}$ is below the diagonal (i.e. the $(k^*, 1)$ domain), the repeated application of $\mathcal{C}$ decreases values without bringing them negative because of Lemma 4.1, so values are eventually all brought to the $[0, k^*]$ domain. There, our derivative is strictly increasing because of convexity, and it stays between 1 and 0.

$$\mathcal{C}'(0) = \frac{(a_1)^2}{\langle \sigma(z)^2 \rangle} \geq 0 \tag{23}$$

$$\mathcal{C}'(k^*) = \frac{1}{1 - k^*} \int_{k^*}^{1} \mathcal{C}'(k^*) dk < \frac{1}{1 - k^*} \int_{k^*}^{1} \mathcal{C}'(k) dk = \frac{1 - k^*}{1 - k^*} = 1 \tag{24}$$

In the $[0, k^*]$ domain, these bounds on the derivative imply bounds on the range.

$$k < k^* - \int_{k}^{k^*} dk < \mathcal{C}(k) = k^* - \int_{k}^{k^*} \mathcal{C}'(j)\, dj \leq k^* \tag{25}$$

Thus, $\mathcal{C}$ is a contracting map in the $[0, k^*]$ domain, and by the Banach fixed-point theorem, has an attractive fixed point reached by repeated application. □

4.3 Proof of Theorem 4.1

Proof. This theorem of course inherits the structure of the overall probability distribution of $P(z^{(\ell)} \mid \mathcal{D})$ from Theorem 3.1. The dataset containing no scalar multiples is equivalent to the condition that initial covariances $K^{(1)}_{\alpha_1\alpha_2}$ are not 1 or -1, and so Lemma 4.3 applies for non-affine activation functions. We will handle affine, nonlinear functions separately. At each layer, the change in covariance is determined by applying the function $\mathcal{C}$, and the covariance approaches its globally attractive fixed point. That fixed point is 0 exactly when $\mathcal{C}(0) = 0$, and from (21), that condition is equivalent to the activation function having zero mean under the Gaussian measure.

$$\mathcal{C}(0) = \frac{(a_0)^2}{\langle \sigma(z)^2 \rangle} = \frac{\langle \sigma(z) \rangle^2}{\langle \sigma(z)^2 \rangle} \tag{26}$$

At a zero fixed point, the derivative must be less than 1 by strict convexity (Lemma 4.1), and so the covariance approaches the fixed point exponentially fast when it is close enough. Thus proving Theorem 4.1 for nonaffine functions.

For affine, nonlinear functions $\sigma(z) = az + b$, the Hermite decomposition is simple enough to compute exactly. From the decomposition we see that their mean under the Gaussian measure is always positive, and repeated applications of $\mathcal{C}$ always brings initial values to a fixed point at 1.

$$\mathcal{C}(k) = \frac{a^2 k + b^2}{\langle \sigma(z)^2 \rangle} \tag{27}$$

Thus, this proves Theorem 4.1 for affine, nonlinear functions. □

5 Conclusion

Under the perturbative expansion and approximation from Roberts et al. (2022), we have established that randomly initialized neural networks behave like random functions and have nearly independent outputs exactly when the activation function σ is nonlinear and has zero mean under the Gaussian measure.

$$\langle \sigma(z) \rangle = 0$$

We now propose these very wide neural networks as a promising alternative construction for the computational no-coincidence conjecture.

The Neural Network Computational No-Coincidence Conjecture: For a randomly initialized neural network $C : \mathbb{R}^n \to \mathbb{R}^n$ with hidden dimension n, depth $\ell = O(\log(n))$, Gaussian random weights, and tanh nonlinearities, let $P(C)$ be the property that for none of the inputs in the dataset $x \in \{-1, 1\}^n$, the output $C(x)$ is all negative. There exists a polynomial-time verification algorithm V that receives as input the neural network C and an advice string π such that for all C where $P(C)$ is true, there exists π with length polynomial in the size of C such that $V(C, \pi) = 1$, and for 99% of random neural networks C, no such π exists.

6 Acknowledgements

We are grateful to Cody Rushing for helpful comments on drafts and to Dmitry Vaintrob for pointing us to Roberts et al. (2022)

References

Bateman, H., & Erdélyi, A. (1953). *Higher transcendental functions* (Vol. 2). New York: McGraw-Hill Book Company. (Based on notes left by Harry Bateman and compiled by the staff of the Bateman Manuscript Project)

Christiano, P., Hilton, J., Lecomte, V., & Xu, M. (2025). Backdoor Defense, Learnability and Obfuscation. In R. Meka (Ed.), *16th innovations in theoretical computer science conference (itcs 2025)* (Vol. 325, pp. 38:1–38:21). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik. Retrieved from `https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2025.38` doi: 10.4230/LIPIcs.ITCS.2025.38

Cloud, A., Goldman-Wetzler, J., Wybitul, E., Miller, J., & Turner, A. M. (2024). *Gradient routing: Masking gradients to localize computation in neural networks.* Retrieved from https://arxiv.org/abs/2410.04332

Gay, W., He, W., Kocurek, N., & O'Donnell, R. (2025). *Pseudorandomness properties of random reversible circuits.* Retrieved from https://arxiv.org/abs/2502.07159

Goldowsky-Dill, N., Chughtai, B., Heimersheim, S., & Hobbhahn, M. (2025). *Detecting strategic deception using linear probes.* Retrieved from https://arxiv.org/abs/2502.03407

Goldwasser, S., Kim, M. P., Vaikuntanathan, V., & Zamir, O. (2024). *Planting undetectable backdoors in machine learning models.* Retrieved from https://arxiv.org/abs/2204.06974

Hanin, B., & Rolnick, D. (2019). *Deep relu networks have surprisingly few activation patterns.* Retrieved from https://arxiv.org/abs/1906.00904

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Delving deep into rectifiers: Surpassing human-level performance on imagenet classification.* Retrieved from https://arxiv.org/abs/1502.01852

Kibble, W. F. (1945). An extension of a theorem of mehler's on hermite polynomials. *Mathematical Proceedings of the Cambridge Philosophical Society*, *41*(1), 12-15.

Lee, J., Bahri, Y., Novak, R., Schoenholz, S. S., Pennington, J., & Sohl-Dickstein, J. (2018). *Deep neural networks as gaussian processes.* Retrieved from https://arxiv.org/abs/1711.00165

Lindsey, J., Gurnee, W., Ameisen, E., Chen, B., Pearce, A., Turner, N. L., ... Batson, J. (2025). On the biology of a large language model. *Transformer Circuits Thread*. Retrieved from https://transformer-circuits.pub/2025/attribution-graphs/biology.html

Neyman, E. (2025). *A computational no-coincidence principle.* Retrieved from https://www.alignment.org/blog/a-computational-no-coincidence-principle/

Palma, G. D., Kiani, B. T., & Lloyd, S. (2019). *Random deep neural networks are biased towards simple functions.* Retrieved from https://arxiv.org/abs/1812.10156

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., & Ganguli, S. (2016). *Exponential expressivity in deep neural networks through transient chaos.* Retrieved from https://arxiv.org/abs/1606.05340

Roberts, D. A., Yaida, S., & Hanin, B. (2022). *The principles of deep learning theory: An effective theory approach to understanding neural networks.* Cambridge University Press. Retrieved from http://dx.doi.org/10.1017/9781009023405 doi: 10.1017/9781009023405

Templeton, A., Conerly, T., Marcus, J., Lindsey, J., Bricken, T., Chen, B., ... Henighan, T. (2024). Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*. Retrieved from https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html

Teney, D., Nicolicioiu, A., Hartmann, V., & Abbasnejad, E. (2024). *Neural redshift: Random networks are not random functions.* Retrieved from https://arxiv.org/abs/2403.02241

Turner, A. M., Thiergart, L., Leech, G., Udell, D., Vazquez, J. J., Mini, U., & MacDiarmid, M. (2024). *Steering language models with activation engineering.* Retrieved from https://arxiv.org/abs/2308.10248

Zou, A., Phan, L., Wang, J., Duenas, D., Lin, M., Andriushchenko, M., ... Hendrycks, D. (2024). *Improving alignment and robustness with circuit breakers.* Retrieved from https://arxiv.org/abs/2406.04313

A Simplified Proof of Theorem 3.1

In this section we will establish the 0th order pertubative expansion of $P(z^{(\ell)} \mid \mathcal{D})$, where we will use big-O notation to hide $O(1/n)$ terms in addition to the $O(1/n^2)$ terms hidden in Theorem 3.1. Since this proof involves heavy use of inverses, we will use raised indices as shorthand for indexing into the inverse of a matrix.

$$\begin{aligned} K^{ij}_{(\ell)} &= ((K^{(\ell)})^{-1})_{ij} \\ \sum_j K^{ij}_{(\ell)} K^{(\ell)}_{jk} &= \delta_{ik} \end{aligned} \tag{28}$$

Here, δ_{ik} is the Kronecker delta.

Proof. This proof will go by induction on the depth of the neural network.

Base case: At the first layer, the distribution of preactivations is exactly Gaussian because the weights and biases, $W^{(1)}_{ij}$ and $b^{(1)}_i$, are a multivariate Gaussian by definition, and each activation $z^{(1)}_{i;\alpha}$ is a linear combination of the weights parameterized by the input x_α. The covariances are as follows:

$$\begin{aligned} \mathbb{E}[z^{(1)}_{i_1;\alpha} z^{(1)}_{i_2;\alpha_2}] &= \mathbb{E}[(b_{i_1} + \sum_j W_{i_1 j}\, x_{j;\alpha_1})(b_{i_2} + \sum_j W_{i_2 j}\, x_{j;\alpha_2})] \\ &= \delta_{i_1 i_2}(C_b + \frac{C_W}{n} \sum_j x_{j;\alpha_1} x_{j;\alpha_2}) \\ &= \delta_{i_1 i_2} K^{(1)}_{\alpha_1 \alpha_2} \end{aligned} \tag{29}$$

Thus, the distribution of first layer preactivations satisfies the inductive hypothesis. Using the notation from (28) for matrix inverses, we can describe the distribution.

$$P(z^{(1)} \mid \mathcal{D}) \propto \exp\left(-\frac{1}{2} \sum_{\alpha_1 \alpha_2 \in \mathcal{D}} K^{\alpha_1 \alpha_2}_{(1)} \sum_{i=1}^{n} z^{(1)}_{i;\alpha_1} z^{(1)}_{i;\alpha_2} \right) \tag{30}$$

Inductive step: We will look at the following recurrence to prove that $P(z^{(\ell)} \mid \mathcal{D})$ is distributed as desired.

$$P(z^{(\ell)} \mid \mathcal{D}) = \int P(z^{(\ell-1)} \mid \mathcal{D}) P(z^{(\ell)} \mid z^{(\ell-1)}) \prod_{i,\alpha} dz^{(\ell-1)}_{i;\alpha} \tag{31}$$

However, before we start working with nearly Gaussian distributions, let's take a brief digression to introduce a useful identity for evaluating expectations. Let $\varepsilon = 1/n$.

$$\begin{aligned} \mathbb{E}[f(z^{(\ell)}_{i_1;\alpha_1}, \ldots, z^{(\ell)}_{i_m;\alpha_m})] &= \int \frac{\exp(-\frac{1}{2}\sum_{i,\alpha} K^{\alpha_1\alpha_2}_{(\ell)} z_{i_1;\alpha_1} z_{i_2;\alpha_2} + O(\varepsilon))}{\left(\int \exp(-\frac{1}{2}\sum_{i,\alpha} K^{\alpha_1\alpha_2}_{(\ell)} z_{i_1;\alpha_1} z_{i_2;\alpha_2} + O(\varepsilon)) \prod_{i,\alpha} dz_{i;\alpha}\right)} f(z) \prod_{i,\alpha} dz_{i;\alpha} \\ &= \int \frac{1}{\sqrt{|2\pi K^{(\ell)}|^n}} \exp\left(-\frac{1}{2} \sum_{i,\alpha} K^{\alpha_1\alpha_2}_{(\ell)} z_{i_1;\alpha_1} z_{i_2;\alpha_2} \right) f(z)(1 + O(\varepsilon)) \end{aligned} \tag{32}$$

In the second equality, we Taylor expanded the expression around $\varepsilon = 0$. We have also dropped the superscript $^{(\ell)}$ to reduce clutter. When the indicies $i_1, \dots i_m$ are equal, we can marginalize over the unused indices and use the notation from (4) for Gaussian expectations.

$$\mathbb{E}[f(z^{(\ell)}_{i;\alpha_1}, \dots, z^{(\ell)}_{i;\alpha_m})] = \langle f(z_{\alpha_1}, \dots, z_{\alpha_m})\rangle_{K^{(\ell)}} + O(\varepsilon) \tag{33}$$

Returning back to (31), the $P(z^{(\ell-1)} \mid \mathcal{D})$ term will be determined by our inductive hypotheses, so let's consider the other term, $P(z^{(\ell)} \mid z^{(\ell-1)})$. Similar to what happened in the first layer with fixed layer inputs $z^{(\ell-1)}$, this distribution is exactly Gaussian.

$$\begin{aligned} P(z^{(\ell)} \mid z^{(\ell-1)}) &= \frac{1}{\sqrt{|2\pi \widehat{G}^{(\ell)}|^n}} \exp\left(-\frac{1}{2}\sum_{i\alpha} \widehat{G}^{\alpha_1\alpha_2}_{(\ell)} z^{(\ell)}_{i_1;\alpha_1} z^{(\ell)}_{i_2;\alpha_2}\right) \\ \widehat{G}^{(\ell)}_{\alpha_1\alpha_2} &= C_b + \frac{C_W}{n}\sum_{i=1}^{n} \sigma^{(\ell-1)}_{i;\alpha_1}\sigma^{(\ell-1)}_{i;\alpha_2} \end{aligned} \tag{34}$$

Here, we have introduced $\widehat{G}^{(\ell)}_{\alpha_1\alpha_2}$, a random variable that depends on $z^{(\ell-1)}$. Let's split $\widehat{G}^{(\ell)}_{\alpha_1\alpha_2}$ into its expectation and deviations.

$$\begin{aligned} G^{(\ell)}_{\alpha_1\alpha_2} &= \mathbb{E}[\widehat{G}^{(\ell)}_{\alpha_1\alpha_2}] = C_b + \frac{C_W}{n}\sum_{i=1}^{n} \mathbb{E}[\sigma^{(\ell-1)}_{i;\alpha_1}\sigma^{(\ell-1)}_{i;\alpha_2}] \\ \widehat{\Delta G}^{(\ell)}_{\alpha_1\alpha_2} &= \widehat{G}^{(\ell)}_{\alpha_1\alpha_2} - G^{(\ell)}_{\alpha_1\alpha_2} \end{aligned} \tag{35}$$

One reason this split will be useful is because the expectation is close to the desired covariance from (9).

$$G^{(\ell)}_{\alpha_1\alpha_2} = K^{(\ell)}_{\alpha_1\alpha_2} + O(\varepsilon) \tag{36}$$

This can be easily proved by induction. Notice that the only difference between the two terms is the expectation $\mathbb{E}[\sigma^{(\ell-1)}_{i;\alpha_1}\sigma^{(\ell-1)}_{i;\alpha_2}]$ instead of the Gaussian expectation $\langle\sigma(z_{\alpha_1})\sigma(z_{\alpha_2})\rangle_{K^{(\ell-1)}}$. At layer 2 when the preactivations are exactly Gaussian, the terms $G^{(2)}_{\alpha_1\alpha_2}$ and $K^{(2)}_{\alpha_1\alpha_2}$ are trivially equal, and in later layers we only need to apply (32) to bound the difference between the nearly-Gaussian and Gaussian expectations. A second reason this split will be useful is because the variance of the deviations is quite small.

$$\mathbb{E}[\widehat{\Delta G}^{(\ell)}_{\alpha_1\alpha_2}\widehat{\Delta G}^{(\ell)}_{\alpha_3\alpha_4}] = \left(\frac{C_W}{n}\right)^2 \sum_{j,k=1}^{n} \mathbb{E}[(\sigma_{j;\alpha_1}\sigma_{j;\alpha_2} - \mathbb{E}[\sigma_{j;\alpha_1}\sigma_{j;\alpha_2}])(\sigma_{k;\alpha_3}\sigma_{k;\alpha_4} - \mathbb{E}[\sigma_{k;\alpha_3}\sigma_{k;\alpha_4}])] \tag{37}$$

In the part of this equation inside the sum, we can use (32) and find that the n^2 off-diagonal terms are $O(\varepsilon)$ and the n diagonal terms are the following:

$$\langle\sigma(z_{\alpha_1})\sigma(z_{\alpha_2})\sigma(z_{\alpha_3})\sigma(z_{\alpha_4})\rangle_{K^{(\ell-1)}} - \langle\sigma(z_{\alpha_1})\sigma(z_{\alpha_2})\rangle_{K^{(\ell-1)}}\langle\sigma(z_{\alpha_3})\sigma(z_{\alpha_4})\rangle_{K^{(\ell-1)}} + O(\varepsilon) \tag{38}$$

These cancel out nicely with the leading coefficient $(1/n)^2$, so the overall variance is $O(\varepsilon)$.

$$\mathbb{E}[\widehat{\Delta G}^{(\ell)}_{\alpha_1\alpha_2}\widehat{\Delta G}^{(\ell)}_{\alpha_3\alpha_4}] = O(\varepsilon) \tag{39}$$

By a similar argument, all the higher moments of $\widehat{\Delta G}$ are also $O(\varepsilon)$. Although the covariances $\widehat{G}^{(\ell)}_{\alpha_1\alpha_2}$ are straightforward to compute, inverting the matrix is slightly harder. We will need the following common identity for inverses of perturbed matrices.

$$\begin{aligned}\widehat{G}^{\alpha_1\alpha_2}_{(\ell)} &= ((G^{(\ell)} + \widehat{\Delta G}^{(\ell)})^{-1})_{\alpha_1\alpha_2} \\ &= G^{\alpha_1\alpha_2}_{(\ell)} - \sum_{\beta} G^{\alpha_1\beta_1}_{(\ell)} \widehat{\Delta G}^{(\ell)}_{\beta_1\beta_2} G^{\beta_2\alpha_2}_{(\ell)} + O(\Delta^2)\end{aligned} \tag{40}$$

At this point we have the tools necessary to expand and simplify (31) to prove our inductive hypothesis. Before doing so however, let's recall the following properties of $\widehat{\Delta G}$ that will be essential for the simplification.

$$\begin{aligned}\mathbb{E}[\widehat{\Delta G}^{(\ell)}_{\alpha_1\alpha_2}] &= 0 \\ \mathbb{E}[O(\Delta^2)] &= O(\varepsilon)\end{aligned} \tag{41}$$

The expansion of (31) will also create really long equations, so let's define the following shorthand.

$$\begin{aligned}S(z) &= -\frac{1}{2}\sum_{i,\alpha} G^{\alpha_1\alpha_2}_{(\ell)} z_{i_1;\alpha_1} z_{i_2;\alpha_2} \\ \Delta H_\alpha &= \frac{1}{2}\sum_{\beta} G^{\alpha_1\beta_1}_{(\ell)} \widehat{\Delta G}^{(\ell)}_{\beta_1\beta_2} G^{\beta_2\alpha_2}_{(\ell)}\end{aligned} \tag{42}$$

Now we are ready to put it all together.

$$
\begin{aligned}
P(z^{(\ell)} \mid \mathcal{D}) &= \int P(z^{(\ell-1)} \mid \mathcal{D}) P(z^{(\ell)} \mid z^{(\ell-1)}) \prod_{i,\alpha} dz_{i;\alpha}^{(\ell-1)} \\
&= \mathbb{E}[P(z^{(\ell)} \mid z^{(\ell-1)})] \\
&= \mathbb{E}\left[\frac{\exp\left(-\frac{1}{2}\sum_{\alpha_1,\alpha_2\in\mathcal{D}} \widehat{G}_{(\ell)}^{\alpha_1\alpha_2} \sum_{i=1}^n z_{i;\alpha_1}^{(\ell)} z_{i;\alpha_2}^{(\ell)}\right)}{\left(\int \exp\left(-\frac{1}{2}\sum_{\alpha_1,\alpha_2\in\mathcal{D}} \widehat{G}_{(\ell)}^{\alpha_1\alpha_2} \sum_{i=1}^n z_{i;\alpha_1}^{(\ell)} z_{i;\alpha_2}^{(\ell)}\right) \prod_{i,\alpha} dz_{i;\alpha}^{(\ell)}\right)} \right] \\
&= \mathbb{E}\left[\frac{\exp\left(S(z) + \sum_{i,\alpha}(\Delta H_\alpha + O(\Delta^2)) z_{i;\alpha_1} z_{i;\alpha_2}\right)}{\left(\int \exp\left(S(z) + \sum_{i,\alpha}(\Delta H_\alpha + O(\Delta^2)) z_{i;\alpha_1} z_{i;\alpha_2}\right) \prod_{i,\alpha} dz_{i;\alpha}\right)} \right] \\
&= \mathbb{E}\Bigg[\frac{\exp(S(z))}{\int \exp(S(z)) \prod_{i,\alpha} dz_{i;\alpha}} \\
&\qquad \times \Bigg(1 + \sum_{i,\alpha} \Delta H_\alpha z_{i;\alpha_1} z_{i;\alpha_2} + O(\Delta^2) \\
&\qquad\qquad - \frac{\int (\sum_\alpha \Delta H_\alpha + O(\Delta^2)) z_{i;\alpha_1} z_{i;\alpha_2} \exp(S(z)) \prod_{i,\alpha} dz_{i;\alpha}}{\int \exp(S(z)) \prod_{i,\alpha} dz_{i;\alpha}} \Bigg) \Bigg] \\
&= \frac{\exp(S(z))}{\int \exp(S(z)) \prod_{i,\alpha} dz_{i;\alpha}^{(\ell)}} (1 + O(\varepsilon)) \\
&\propto \exp(S(z) + O(\varepsilon)) \\
&\propto \exp\left(-\frac{1}{2} \sum_{i,\alpha} K_{(\ell)}^{\alpha_1\alpha_2} z_{i;\alpha_1}^{(\ell)} z_{i;\alpha_2}^{(\ell)} + O(\varepsilon) \right)
\end{aligned}
\tag{43}
$$

In the fourth equality we inserted (40) with notation from (42), in the fifth we Taylor expanded around $\varepsilon = 0$, in the sixth we applied (41), the useful properties of $\widehat{\Delta G}$, in the seventh we took the logorithm and Taylor expanded to move the $O(\varepsilon)$ term into the exponent, and in the last equality we applied (36), the difference bound between $\widehat{G}_{\alpha_1\alpha_2}^{(\ell)}$ and $K_{\alpha_1\alpha_2}^{(\ell)}$, combined with (40) to bound the difference of their inverses. Now that we have proven that the probability distribution is of the desired form, it's simple to check that the covariances are as desired as well. We can even use (32) again.

$$
\begin{aligned}
\mathbb{E}[z_{i_1;\alpha_1}^{(\ell)} z_{i_2;\alpha_2}^{(\ell)}] &= \delta_{i_1 i_2} \langle z_{\alpha_1}^{(\ell)} z_{\alpha_2}^{(\ell)} \rangle_{K^{(\ell)}} + O(\varepsilon) \\
&= \delta_{i_1 i_2} K_{\alpha_1\alpha_2}^{(\ell)} + O(\varepsilon)
\end{aligned}
\tag{44}
$$

Thus, we have completed the proof that the probability distribution $P(z^{(\ell)} \mid \mathcal{D})$ and its covariances are of the desired form. □

B Derivation of Equation 20

Let $\gamma_\Sigma(u_1, u_2)$ and $\gamma_I(u_1, u_2)$ represent measures on the 2D unit-variance Gaussian when the covariance is k and 0.

$$\begin{aligned}\gamma_\Sigma(u_1, u_2) &= \frac{1}{2\pi\sqrt{1-k^2}} \exp\left(\frac{2u_1u_2k - u_1^2 - u_2^2}{2(1-k^2)}\right) \\ \gamma_I(u_1, u_2) &= \frac{1}{2\pi} \exp\left(\frac{-u_1^2 - u_2^2}{2}\right)\end{aligned} \tag{45}$$

The following is a standard formulation of Mehler's equation in terms of Gaussian measures (Kibble, 1945).

$$\gamma_I(u_1, u_2) \sum_{n=0}^{\infty} \frac{k^n}{n!} \mathrm{He}_n(u_1)\mathrm{He}_n(u_2) = \gamma_\Sigma(u_1, u_2) \tag{46}$$

Using Mehler's equation, we can prove (20), the product of two Hermite polynomials under correlated Gaussian weighting.

$$\begin{aligned}&\langle \mathrm{He}_n(z_1)\mathrm{He}_m(z_2)\rangle_\Sigma, \quad \Sigma = \begin{pmatrix} 1 & k \\ k & 1 \end{pmatrix} \\ =& \iint \mathrm{He}_n(z_1)\mathrm{He}_m(z_2)\gamma_\Sigma(z_1, z_2)\, dz_1\, dz_2 \\ =& \iint \mathrm{He}_n(z_1)\mathrm{He}_m(z_2) \left(\sum_{i=0}^{\infty} \frac{k^i}{i!}\mathrm{He}_i(z_1)\mathrm{He}_i(z_2)\right) \gamma_I(z_1, z_2)\, dz_1\, dz_2 \\ =& \left\langle \mathrm{He}_n(z_1)\mathrm{He}_m(z_2) \left(\sum_{i=0}^{\infty} \frac{k^i}{i!}\mathrm{He}_i(z_1)\mathrm{He}_i(z_2)\right)\right\rangle \\ =& \left\langle \mathrm{He}_m(z_2) \left(\sum_{i=0}^{\infty} \left\langle \frac{k^i}{i!}\mathrm{He}_n(z_1)\mathrm{He}_i(z_1)\right\rangle \mathrm{He}_i(z_2)\right)\right\rangle \\ =& \left\langle \mathrm{He}_m(z_2) \left\langle \frac{k^n}{n!}\mathrm{He}_n(z_1)^2\right\rangle \mathrm{He}_n(z_2)\right\rangle \\ =& \langle \mathrm{He}_m(z_2)\mathrm{He}_n(z_2)k^n\rangle \\ =& \,\delta_{nm}\, n!\, k^n\end{aligned} \tag{47}$$