On the impact of activation and normalization in obtaining isometric embeddings at initialization

Amir Joudaki
ETH Zurich
amir.joudaki@inf.ethz.ch

Hadi Daneshmand
MIT/LIDS-FODSI
dhadi@mit.edu

Francis Bach
INRIA-ENS-PSL Paris
francis.bach@inria.fr

Abstract

In this paper, we explore the structure of the penultimate Gram matrix in deep neural networks, which contains the pairwise inner products of outputs corresponding to a batch of inputs. In several architectures it has been observed that this Gram matrix becomes degenerate with depth at initialization, which dramatically slows training. Normalization layers, such as batch or layer normalization, play a pivotal role in preventing the rank collapse issue. Despite promising advances, the existing theoretical results do not extend to layer normalization, which is widely used in transformers, and can not quantitatively characterize the role of non-linear activations. To bridge this gap, we prove that layer normalization, in conjunction with activation layers, biases the Gram matrix of a multilayer perceptron towards the identity matrix at an exponential rate with depth at initialization. We quantify this rate using the Hermite expansion of the activation function.

1 Introduction

Optimization of deep neural networks is a challenging non-convex problem. Various components and optimization techniques are developed over five decades to make the optimization feasible. Components such as activation functions [Hendrycks and Gimpel, 2016], normalization layers [Ioffe and Szegedy, 2015], and residual connections [He et al., 2016] have significantly influenced network training, hence become the building blocks of neural networks. For example, the training of large language models hinges on the careful utilization of residual connections, normalization layers, and tailored activations [Vaswani et al., 2017, Radford et al., 2018]. Noci et al. [2022a] highlight that the absence or improper utilization of these components can substantially slows training. The practical success of these components has inspired extensive theoretical studies on the intricate interplay between optimization and building blocks of deep neural networks [Saxe et al., 2013, Daneshmand et al., 2020, 2021, 2023, Joudaki et al., 2023, Pennington et al., 2018, Kohler et al., 2019].

To understand the influence of these building blocks on training, a line of research studies neural networks at initialization. In particular, several studies focus on the penultimate Gram matrix [Yang et al., 2019, Pennington et al., 2018, de G. Matthews et al., 2018, Li et al., 2022, Jacot et al., 2018], which contains the inner product of outputs for a batch of inputs. These studies have revealed that as the network depth increases, The Gram matrix can become degenerate at initialization in multilayer perceptrons (MLPs) [Saxe et al., 2013, Daneshmand et al., 2020], convolutional networks [Bjorck et al., 2018], and transformers [Dong et al., 2021]. The degenerate Gram matrix can, in turn, slow down the training process [Noci et al., 2022b, Pennington et al., 2018, Xiao et al., 2018]. Further studies have shown that normalization layers are an effective tool to avoid Gram degeneracy, thereby enhancing training [Yang et al., 2019, Daneshmand et al., 2020, 2021, Bjorck et al., 2018].

Despite the tremendous advance, a gap between theory and practice persists. Many theoretical studies rely on the regime of infinite depth [Yang et al., 2019, Pennington et al., 2018], or technical assumptions that are hard to validate [Daneshmand et al., 2021, 2020, Joudaki et al., 2023]. Furthermore,

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

existing results are limited to initialization where the parameters are random. Hence, these results do not necessarily hold during or after training. In the present work, we set out to bridge these gaps.

Contributions. We introduce the notion of isometry, which quantifies the similarity of the Gram matrix to the identity. Our first theoretical result is that isometry is non-decreasing under (batch and layer) normalization. This result reveals that normalization layers bias intermediate representations towards isometry at initialization, during, and after training.

We further analyze the influence of non-linear activations on the isometry of intermediate representations in MLPs. In the mean-field regime, we prove non-linear activations bias the intermediate representations towards isometry at an exponential rate in depth. Our main contribution is *quantifying the rate with Hermit polynomial expansion of activations*. Surprisingly, our experiments show that this rate correlates with the convergence of stochastic gradient descent in MLPs with standard non-linear activation functions in practice.

2 Related works

A line of research investigates the interplay between neural architectures and training. The existing literature postulates that in order to ensure fast training [Schoenholz et al., 2017, Poole et al., 2016], the network output must be sensitive to input changes, quantified by the spectrum of input-input Jacobean. This hypothesis is employed by Xiao et al. [2018] to train a 10,000-layer CNN using proper weight initialization without stabilizing components such as skip connection or normalization layers. He et al. [2023] demonstrate the critical role of the Jacobean spectra in large language models. In this paper, we analyze the spectrum of Gram matrices that connect to the spectral properties of input-output Jacobean.

Mean-field theory has been extensively used to characterize the effect of neural architectures on the Gram matrix in the limit of infinite depth and width. In this setting, the Gram matrix is a fixed point of a recurrence equation that depends on the neural architecture [Schoenholz et al., 2017, Yang et al., 2019, Pennington et al., 2018]. A fixed-point analysis can provide insights into the structure of Gram matrices in deep neural networks, thereby shedding light on the degeneracy of Gram matrices in networks without normalization [Schoenholz et al., 2017, Yang et al., 2019]. However, often fixed-points are not unique, and they can be degenerate or non-degenerate Yang et al. [2019]. In this paper, we establish a convergence rate to a non-degenerate fixed-point for a family of MLPs.

Batch normalization [Ioffe and Szegedy, 2015] and layer normalization [Ba et al., 2016] layers are widely used in deep neural networks (DNNs) to improve training. Batch normalization ensures that each feature within a layer across a mini-batch has zero mean and unit variance. In contrast, layer normalization centers and divides the output of each layer by its standard deviation. While focusing on layer normalization, we characterize a property shared between batch and layer normalization.

A broad spectrum of activation functions such as ReLU [Fukushima, 1969], GeLU [Hendrycks and Gimpel, 2016], SeLU [Klambauer et al., 2017], and Hyperbolic Tangent, and Sigmoid, are used in DNNs. These functions have various computational and statistical consequences in deep learning. Despite this diversity, only the design of SeLU activation is theoretically motivated [Klambauer et al., 2017], while a broader theoretical understanding of activations remains elusive. To address this issue, we develop a theoretical framework to characterize influence of a broad range of activations on intermediate representations in DNNs.

3 Preliminaries

Notation. Let $\langle x, y\rangle$ be the inner product of vectors x and y, and $\|x\|^2 = \langle x, x\rangle$ the squared Euclidean norm of x. For a matrix X, we write $X_{i\cdot}$ and $X_{\cdot i}$ for the i-th row and column of X, respectively. We use $W \sim N(\mu, \sigma^2)^{m\times n}$ to indicate that W is an $m \times n$ Gaussian matrix with i.i.d. elements from $N(\mu, \sigma^2)$. We denote by $\mathbf{0}_n$ the zero vector of size n. Given vector $x \in \mathbb{R}^n$, $\overline{x}$ denotes the arithmetic mean of $\frac{1}{n}\sum_{i=1}^n x_i$. Lastly, I_n is the identity matrix of size n.

Normalization layers. Let $\textsc{Ln} : \mathbb{R}^d \to \mathbb{R}^d$ and $\textsc{Bn} : \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n}$, denote batch normalization and layer normalization respectively. Table 1 summarizes the definition of normalization layers. In

our notations, we separate centering from normalization in layer (batch) normalization. Similarly, we split batch normalization into centering and normalization steps in our definitions. This notation allows us to decouple the effect of normalization from the centering. However, we will not depart from the standard MLP architectures as we include centering in the network architecture defined below.

Width	$d \in \mathbb{N}$	Batch size	$n \in \mathbb{N}$
Depth	$L \in \mathbb{N}$	Input	$x \in \mathbb{R}^d$
Input batch	$X \in \mathbb{R}^{d \times n}$	Gaussian weights	$W^1, \dots, W^L \sim N(0,1)^{d \times d}$
Activation	$\sigma : \mathbb{R} \to \mathbb{R}$	Centering	$x - \overline{x}$
Layer Norm	$\textsc{Ln}(x) = \frac{x}{\sqrt{\frac{1}{d}\sum_i^d x_i^2}}$	**Batch Norm**	$\textsc{Bn}(X)_{ij} = \frac{X_{ij}}{\sqrt{\frac{1}{n}\sum_k^d X_{ik}^2}}$

Table 1: Building blocks we consider in this work.

MLP setup. The subject of our analysis is an MLP with constant width d across the layers and L layers, which takes input $x \in \mathbb{R}^d$ and maps it to output $x^L \in \mathbb{R}^d$, with hidden representations as

$$\begin{cases} x^{\ell+1} = \frac{1}{\sqrt{d}}\textsc{Ln}(h_\ell - \overline{h}_\ell), \quad h_\ell = \sigma(W^\ell x^\ell), & \ell = 0, \dots, L-1 \\ x^0 := \frac{1}{\sqrt{d}}\textsc{Ln}(x - \overline{x}), & \text{input.} \end{cases} \tag{1}$$

While the original ordering of layer normalization and activation is different [Ba et al., 2016], Xiong et al. [2020] show that the above ordering is more effective for large language models.

Gram matrices and isometry. Given n data points $\{x_i\}_{i \le n} \in \mathbb{R}^d$, the Gram matrix G^ℓ of the feature vectors $x_1^\ell, \dots, x_n^\ell \in \mathbb{R}^d$ at layer ℓ of the network is defined as

$$G^\ell := \left[\langle x_i^\ell, x_j^\ell \rangle\right]_{i,j \le n}, \qquad \ell = 0, 1, \dots, L. \tag{2}$$

We define the notion of isometry to measure how much G^ℓ is close to a scaling factor of the identity matrix.

Definition 1. *Let G be an $n \times n$ positive semi-definite matrix. We define the* isometry $\mathcal{I}(G)$ *of G as the ratio of its normalized determinant to its normalized trace:*

$$\mathcal{I}(G) := \frac{\det(G)^{1/n}}{\frac{1}{n}\mathrm{tr}(G)}. \tag{3}$$

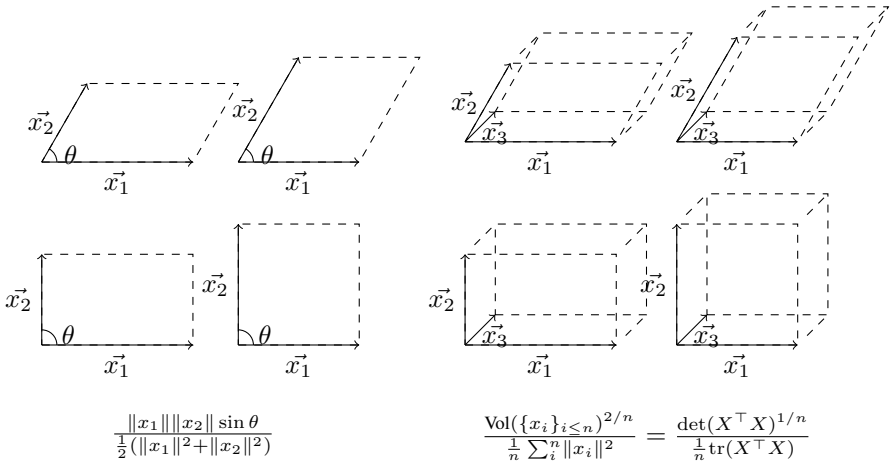


Figure 1: A geometric interpretation of isometry: higher volume, in the second row, corresponds to higher value for isometry.

$\mathcal{I}(G^\ell)$ is a scale-invariant quantity measuring the parallelepiped volume spanned by the feature vectors $x_1^\ell, \ldots, x_n^\ell$. For example, consider two points on a plane $x_1, x_2 \in \mathbb{R}^2$ with lengths $a = |x_1|, b = |x_2|$ and angle $\theta = \angle(x_1, x_2)$. The ratio is given by $ab\sin(\theta)/(a^2 + b^2)$, which is maximized when $a = b$ and $\theta = \pi/2$. This relationship between volume and isometry is visually clear $n = 2$ and $n = 3$ feature vectors in Figure 1.

Remarkably, $\mathcal{I}(M)$ has the following properties (see Lemma A.1 for formal statements and proofs):

(i) *Scaling-invariant:* For all constants $c > 0$, we have $\mathcal{I}(G) = \mathcal{I}(cG)$.

(ii) *Range:* $\mathcal{I} \in [0, 1]$ where the boundaries 0 and 1 are achieved for to degenerate and identity matrices respectively.

We leverage (ii) to introduce the **isometry gap** defined as $-\log \mathcal{I}(M)$ lies between 0 and ∞, with 0 and ∞ indicating the identity and degenerate matrices respectively. Isometry allows us to establish the inherent bias of normalization layers in the following section

4 Isometry bias of normalization

This notion of isometry has a remarkable property: if we normalize each point by its Euclidean norm, then the isometry of their associated Gram matrix does not decrease. We formalize this property in the following theorem.

Theorem 1. *Given n samples $\{x_i\}_{i\le n} \subset \mathbb{R}^d \setminus \{\mathbf{0}_d\}$, and their projection onto the unit sphere $\widetilde{x}_i := x_i/\|x_i\|$, and their respective Gram matrices $G = \left[\langle x_i, x_j\rangle\right]_{i,j\le n}$ and $\widetilde{G} = \left[\langle \widetilde{x}_i, \widetilde{x}_j\rangle\right]_{i,j\le n}$. The isometry of Gram matrices obeys*

$$\mathcal{I}\left(\widetilde{G}\right) \ge \mathcal{I}(G)\left(1 + \frac{\frac{1}{n}\sum_i^n (a_i - \bar{a})^2}{\bar{a}^2}\right), \qquad \text{where } a_i := \|x_i\|, \bar{a} := \frac{1}{n}\sum_i^n a_i. \tag{4}$$

Geometric interpretation of Theorem 1. Recall that isometry can be interpreted as a scaling(and dimension)-free notion of volume. If $x_1, \ldots, x_n$ denote columns of X, determinant is the squared volume of this parallel-piped spanned by $x_1, \ldots, x_n$, and trace is the sum of Euclidean norm $\sum_i^n \|x_i\|^2$. Thus, projection onto the unit sphere will only change the lengths of these samples while leaving the angles intact.

Theorem 1 has an interesting geometric implication: Among parallel-piped shapes with similar angles between their edges, the one with equal edge lengths has the highest isometry. Theorem 1 further shows a subtle property of normalization: as long as there is some variation in the sample norms, i.e., $\|x_i\|$'s are not all equal, the post-normalization Gram has strictly higher isometry than the pre-normalization Gram matrix. It further quantifies the improvement in isometry as a function of variation of norms. Intuitively, terms $\bar{a}$ and $\frac{1}{n}\sum_i^n (a_i - \bar{a})^2$ can be interpreted as the average and variance of sample norms $a_1, \ldots, a_n$. Thus, there is a higher variation in the norms more isometry after normalization.

The high-level intuition behind the proof is that we decouple the role of angles and edge lengths in volume formulation, which is evident in two dimensions in Figure 1. Furthermore, by the arithmetic vs geometric mean inequality, it is clear that the ratio is maximized when the norms are all equal. Remarkably, the proof for the general case is as simple as the two-dimensional case.

Proof of Theorem 1. Define $D := \mathrm{diag}(a_1/\sqrt{d}, \ldots, a_n/\sqrt{d})$. Observe that $C = D\widetilde{G}D$, implying $\det(G) = \det(\widetilde{G})\det(D)^2$. Because $\widetilde{x}_i$'s have norm $\sqrt{d}$, diagonals of Gram after normalization are

constant $\widetilde{G}_{ii} = d$, implying $\frac{1}{n}\mathrm{tr}(\widetilde{G}) = d$. We have

$$\frac{\mathcal{I}(\widetilde{G})}{\mathcal{I}(G)} = \frac{\frac{1}{n}\mathrm{tr}(G)}{\frac{1}{n}\mathrm{tr}(\widetilde{G})}\frac{\det(\widetilde{G})^{1/n}}{\det(G)^{1/n}} \tag{5}$$

$$= \frac{\frac{1}{n}\sum_i^n a_i^2}{d}\frac{\det(\widetilde{G})^{1/n}}{\det(\widetilde{G})^{1/n}(d^{-n}\prod_i^n a_i^2)^{1/n}} \tag{6}$$

$$= \frac{(\frac{1}{n}\sum_i a_i)^2}{(\prod_i^n a_i)^{2/n}}\frac{\frac{1}{d}\sum_i^n a_i^2}{(\frac{1}{n}\sum_i a_i)^2} \tag{7}$$

$$= 1 + \frac{\frac{1}{n}\sum_i^n (a_i - \bar{a})^2}{\bar{a}^2}, \qquad \bar{a} := \frac{1}{n}\sum_i^n a_i \tag{8}$$

□

4.1 Implications for layer (and batch) normalization

Theorem 1 allows us to characterize the isometry bias of layer and batch normalization.

Corollary 2. *Consider n vectors before and after layer-normalization $\{x_i\}_{i\leq n} \subset \mathbb{R}^d \setminus \{\mathbf{0}_d\}$ and $\{\widetilde{x}_i\}_{i\leq n}, \widetilde{x}_i := \textsc{Ln}(x_i)$. Define their respective Gram matrices $G := [\langle x_i, x_j\rangle]_{i,j\leq n}$, and $\widetilde{G} := [\langle \widetilde{x}_i, \widetilde{x}_j\rangle]_{i,j\leq n}$. We have:*

$$\mathcal{I}\left(\widetilde{G}\right) \geq \mathcal{I}(G)\left(1 + \frac{\frac{1}{n}\sum_i^n (a_i - \bar{a})^2}{\bar{a}^2}\right), \qquad \text{where } a_i := \|x_i\|, \bar{a} := \frac{1}{n}\sum_i^n a_i.$$

What makes the above result apart from related studies [Daneshmand et al., 2021, 2020, Yang et al., 2019] is that the increase in isometry is not limited to random initialization. Thus, layer normalization increases the isometry even during and after training. This calls for future research on the role of this inherent bias in enhanced optimization and generalization performance with batch normalization [Ioffe and Szegedy, 2015, Yang et al., 2019, Lyu et al., 2022, Kohler et al., 2019].

Despite the seemingly vast differences between layer normalization and batch normalization [Lubana et al., 2021], the following corollary shows a link between these two different normalization techniques.

Corollary 3. *Given n samples in a mini-batch before $X \in \mathbb{R}^{d\times n}$, and after normalization $\widetilde{X} = \textsc{Bn}(X)$ and define covariance matrices $C := XX^\top$ and $\widetilde{C} := XX^\top$. We have:*

$$\mathcal{I}\left(\widetilde{C}\right) \geq \mathcal{I}(C)\left(1 + \frac{\frac{1}{d}\sum_i^d (a_i - \bar{a})^2}{\bar{a}^2}\right), \qquad \text{where } a_i := \|X_{i\cdot}\|, \bar{a} := \frac{1}{n}\sum_{i=1}^d a_i.$$

Gram matrices of networks with batch normalization have been the subject of many previous studies at network initialization: it has been postulated that BN prevents rank collapse issue [Daneshmand et al., 2020] and that it orthogonalizes the representations [Daneshmand et al., 2021], and that it imposes isometry [Yang et al., 2019]. It is straightforward to verify that orthogonal matrices have the maximum isometry. Thus, the increase in isometry links to the orthogonalization of hidden representation charaterized by Daneshmand et al. [2021]. While all previous results heavily rely on Gaussian random weights to establish this inherent bias, Corollary 3 is not limited to random weights.

4.2 Empirical validation in an MLP setup

Figure 2 shows the isometry gap of various layers in an MLPs with layer normalization and various activation functions at initialization (left) and after training on `CIFAR10` training set. Shades in the figure mark layers illustrate that the isometry of the Gram matrix is non-increasing after each layer normalization layer. We can see in Figure 2 that both before (left) and after training (right), the normalization layers maintain or improve isometry. This can be seen visually because the isometry gap in the normalization layers (shaded blue) is either stable or declining.

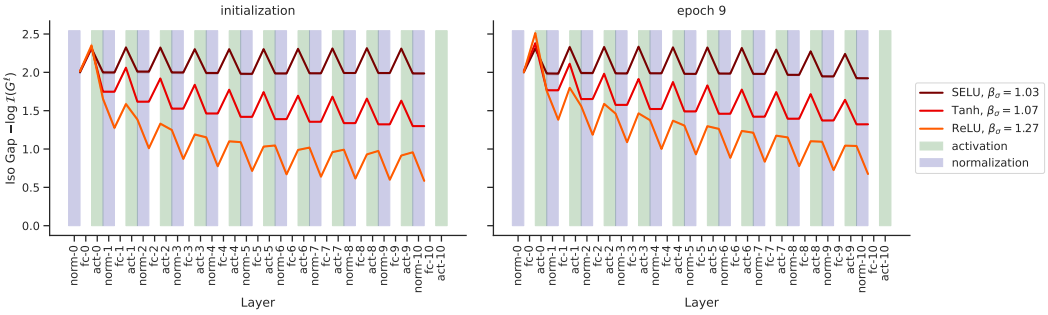


Figure 2: **Validation of isometry bias of normalization and activations** *Validation of Corollary 2*: Normalization layers (shaded blue) across all layers and configurations maintain or decrease isometry gap both before (left) and after (right) training. *Validation of Theorem 4:* The higher the isometry strength β_σ, the faster the decay of isometry gap. Activation layers (shaded green) decrease the isometry gap. *Hyper parameters:* depth: 10, width: 1000, batch-size: 512, training SGD on training set of `CIFAR10`. with $lr = 0.01$.. Layer names are encoded as type-index, where type can be fc: fully connected, norm: LayerNorm, or act: activation.

5 Isometry and non-linear activation functions

So far, we have studied the influence of individual normalization layers. Now, we analyze the dynamic of data across all layers of MLPs. Our analysis is based on the notion of isometry. Assuming the weights are random, we characterize how $\mathcal{I}(G^\ell)$ changes with ℓ. Specifically, we investigate how non-linear activations influence the isometry, shaping data across the layers.

5.1 Hermite expansion of activation functions

It is hard to analyze the dynamic of Gram matrices after applying non-linear activations. To tackle this theoretical challenge, we leverage Hermite polynomial expansion of activations. Hermite polynomials have been successfully used to analyze the kernel function associated with neural networks at initialization [Daniely et al., 2016] and to derive neural tangent kernel derivations of infinitely wide networks [Yang, 2019]. Taking inspiration from these studies, we characterize the influence of activations on isometry.

Definition 2. *Hermite polynomials of degree k denoted by $He_k(x)$ is defined as*

$$He_k(x) := (k!)^{-\frac{1}{2}}(-1)^k e^{\frac{x^2}{2}} \frac{d^k}{dx^k} e^{-\frac{x^2}{2}}.$$

Hermite polynomials possess the properties of completeness and orthogonality under the Gaussian weight kernel. This means that any function in the space of square-integrable functions with respect to the Gaussian kernel, satisfying the condition $\int_{-\infty}^{\infty} \sigma(x)^2 e^{-x^2/2} dx < \infty$, can be expressed as a linear combination of Hermite polynomials. For a given scalar function σ, define its normalized Hermit coefficients c_k is defined as (See section B for more details):

$$c_k := \mathbb{E}_{x\sim\mathcal{N}(0,1)}[\sigma(x) He_k(x)].$$

To quantify the influence of activations on the isometry, we define the notion of isometry strength for activations.

Definition 3 (Isometry strength)**.** *Given activation σ with Hermite expansion $\{c_k\}_{k\geq 0}$, define its* isometry strength β_σ *as:*

$$\beta_\sigma := 2 - \frac{c_1^2}{\sum_{k=1}^{\infty} c_k^2} \tag{9}$$

Table 2 presents the isometry strength of various activations in closed form.

5.2 Mean-field Isometry for non-linear activations

In this section, we analyze how $\mathcal{I}(G^\ell)$ changes with ℓ. We use the mean-field dynamic of Gram matrices subject of previous studies [Yang et al., 2019, Schoenholz et al., 2017, Poole et al., 2016].

	$He_1(x)$	$He_2(x)$	Sine	Exponential	Step	ReLU
σ	x	x^2-1	$\sin(x)$	$\exp(x-2)$	$\mathbf{1}[x>0]$	$\max(x,0)$
β_σ	1	2	$2-\frac{2e}{e^2-1}$	$2-\frac{1}{e-1}$	$2-\frac{2}{\pi}$	$\frac{3\pi-4}{2\pi-2}$

Table 2: Isometry strength β_σ (see Definition 3) for various activation functions.

The mean-field dynamics of Gram matrices is given by

$$G_*^{\ell+1} = \mathbb{E}_{h\sim N(0,G_*^\ell)}\left[\phi(\sigma(h))\phi(\sigma(h))^\top\right], \qquad \text{where } [\phi(a)]_i := (a_i - \mathbb{E}a_i)/\sqrt{\operatorname{Var} a_i}. \tag{10}$$

This equation represents the expected Gram matrix of layer $\ell+1$, given the Gram matrix of the previous layer is G_*^ℓ, with ϕ simulates layer normalization in the mean-field regime (see section B for more details). The sequence G_*^ℓ approximates the dynamics of G^ℓ, and this correspondence becomes exact for infinitely wide MLPs. In the rest of this section, we analyze the above dynamical system.

Inspired by the isometry bias of normalization layers, we analyze how the isometry of G_*^ℓ changes with depth. Interestingly, the negative log of isometry $-\log \mathcal{I}(G_*^\ell)$ can serve as a Lyapunov function for the above dynamics. The following theorem proves non-linear activations also impose isometry similar to normalization layers.

Theorem 4. *Let σ be an activation function with a Hermite expansion and a isometry strength β_σ, (see equation (9)). Given the input samples are not aligned, there is a constant $C>0$ such that we have*

$$-\log \mathcal{I}(G_*^\ell) \le Cn\beta_\sigma^{-\ell}. \tag{11}$$

Note that the condition on input samples not duplicated is essential to reach isometry through depth. For example, if the input batch contains a duplicated sample, their corresponding representations across all layers will remain duplicated, implying that all G^ℓ's will be degenerate.

Thus, MLPs with normalization layers and non-linear activations can achieve isometric embedding by increasing depth at initialization. This reveals the importance of $\beta_\sigma > 1$, hence non-linear activations in isometry. This constant can be computed in closed form for various activations, as shown in Table 2. Figure 3 compares the established bound on the isometry gap with those observed in practice, i.e. G^ℓ, for three activations. We observe β_σ predicts the decay rate in isometry of Gram matrices G^ℓ. We will experimentally validate the link between isometry strength and training.

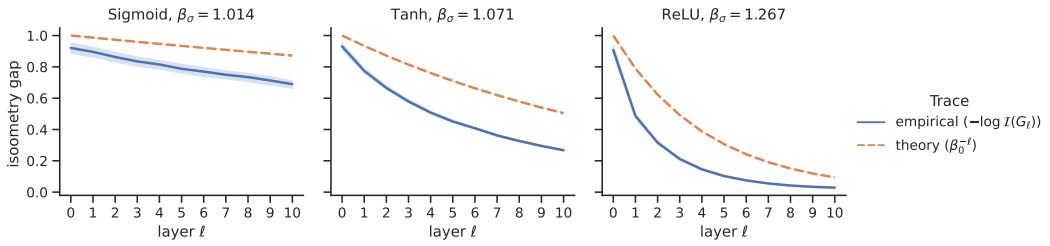


Figure 3: Isometry gap vs. depth ℓ. Solid blue traces show the isometry of MLP with $n = 100$, $d = 5000$, and various activations σ. Solid lines show the average of $\#10$ independent runs. The dashed traces are theoretical upper bounds given in Theorem 4, with constant $C = 1$.

6 Implications of our theory

We first provide insights on layer normalization through Hermit expansion of activation; then, we show the implications of our analysis on training. Layer normalization has two main components: (i) centering and (ii) normalizing the norms. We show how these components can be explained via Hermit expansion of activation functions. To ensure isometric embedding, we propose alternatives to centering and normalization based on insights from Hermit expansion of activation in Sections 6.1 and 6.2, respectively. Finally, we conclude with an experimental result showing a correlation between isometry strength and training with SGD in shallow MLPs with standard activation functions.

Experimental Setup. Let a^ℓ refers to the post-activation vector $a^\ell := \sigma(W^\ell x^\ell), W^\ell \sim N(0, 1/d)^{d\times d}$. Compared to equation (1), the weight matrix has absorbed the factor $1/\sqrt{d}$, and thus its elements are drawn from $N(0, 1/d)$ instead of $N(0, 1)$. Note that the offset $c_0 = \mathbb{E}_{z\sim N(0,1)}\sigma(z)$, and scale $\bar{\sigma}(1) := \mathbb{E}_{z\sim N(0,1)}(\sigma(z) - c_0)^2 = \sum_{k=1}^{\infty} c_k^2$, are absolute constants given the activation.

For training, the task is image classification for CIFAR10 [Krizhevsky et al.]. The model used for classification is MLPs with layer normalization and various activation functions. Following our theoretical settings, we use Gaussian random weight matrices. This has a constant width of 1000 across the hidden layers of size 1000 with 10 layers. We use stochastic gradient descent (SGD) with a constant stepsize 0.01 and batch size 512 to train the network.

6.1 Centering and Hermit expansion

The established isometry bias in Theorem 4 relies on the centering of coordinates, defined in Table 1. We experimentally show that the centering can be replaced by a slight modification of the activation function: zeroing out the c_0 term in the Hermite expansion. We explicitly remove the offset term c_0 from the activation function instead of centering as $\sigma(x) - c_0$. Strikingly, our experiments presented in Figure 4 indicate that this modification of activation can effectively replace centering in layer normalization. This result provides novel insights into the role of centering in layer normalization.

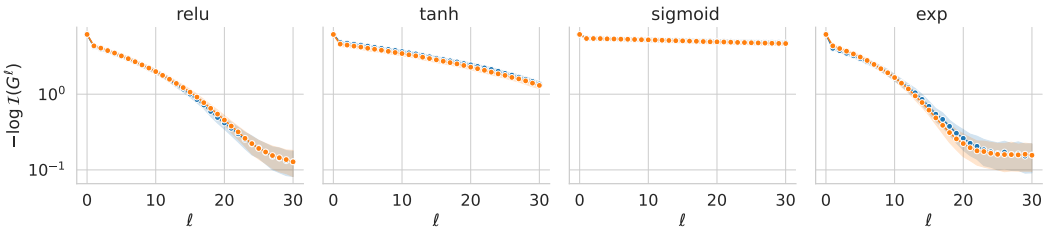


Figure 4: Layer vs. mean-field centering in obtaining isometry. Batch size $n = 10$, width $d = 1000$. Layer centering (orange): $x^{\ell+1} = \textsc{Ln}(a^\ell - \overline{a^\ell})$. Mean-field centering (blue): $x^{\ell+1} = \textsc{Ln}(a^\ell - c_0)$.

6.2 Normalization and Hermit expansion

Theorem 4 requires the normalization of norms in addition to centering to achieve isometry. Figure 5 underlines the importance of this projection for different activations where we observe the isometry gap may increase without normalization.

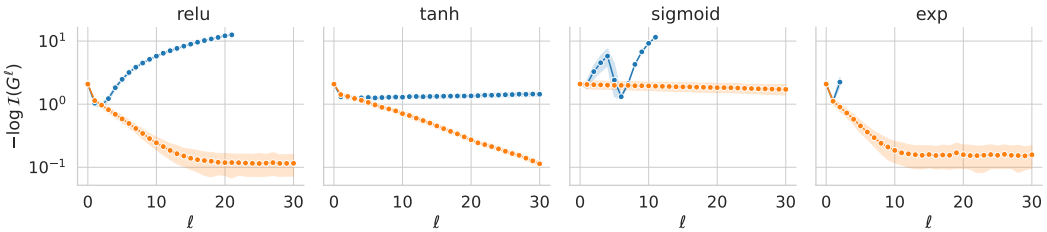


Figure 5: Importance of projection in obtaining isometry, batch size $n = 10$, width $d = 1000$. No projection (blue): $x^{\ell+1} = (a^\ell - c_0)$. Layer projection (orange): $x^{\ell+1} = \textsc{Ln}(a^\ell - c_0)$.

According to our mean-field analysis, the factor $\frac{1}{d}\sum_{i=1}^{d}(a_i^\ell - \overline{a^\ell})^2$ in layer normalization converges to variance $\text{var}_{z\sim N(0,1)}(\sigma(z)) = \bar{\sigma}(1) = \sum_{k=1}^{\infty} c_k^2$. Thus, as the width increases, the layer normalization operator $\textsc{Ln}(a^\ell - \overline{a^\ell})$ will converge to $(a^\ell - c_0)/\sqrt{\bar{\sigma}(1)}$. Figure 6 demonstrates that the constant scaling $1/\sqrt{\bar{\sigma}(1)}$, achieves comparable isometry to layer projection for hyperbolic tangent and sigmoid and ReLU, while it is not effective for exp function. This observation calls for future research on the link between normalization and activation in deep neural networks.

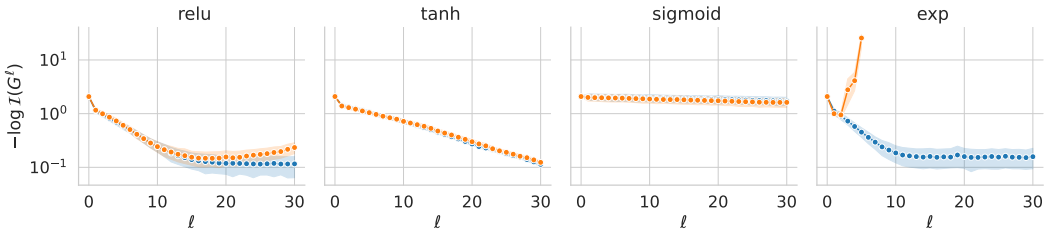


Figure 6: Layer vs mean-field projections. Batch size $n = 10$, width $d = 1000$.
Layer projection (blue): $x^{\ell+1} = \textsc{Ln}(a^\ell - c_0)$. Mean-field projection (orange): $x^{\ell+1} = \frac{a^\ell - c_0}{\sqrt{\bar{\sigma}(1)}}$.

6.3 Isometry strength correlates with SGD convergence rate in shallow MLPs.

We observe that the convergence of SGD correlates with isometry strength in a specific range of neural network hyper-parameters. Figure 7 shows the convergence of SGD is faster for activations with a significantly larger isometry strength (see Definition 3). While striking, we highlight two important limitations:

- The correlation between isometry strength and convergence rate only holds for *shallow* MLPs, and we did not observe the same correlation for deeper networks. This may be due to the issue of gradient explosion studied by Meterez et al. [2023].
- Remarkably, the isometry strength is not the only factor influencing training. Specifically, the convergence heavily depends on the *expressive power* of non-linear activations for function approximation. Thus, we only compare the correlation for standard activations used in deep neural networks.

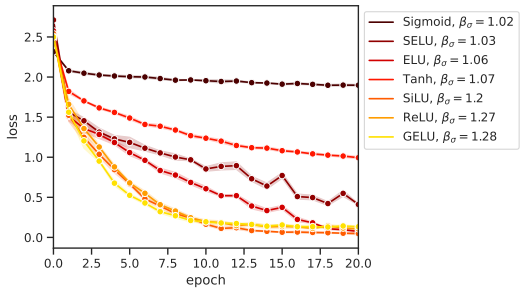
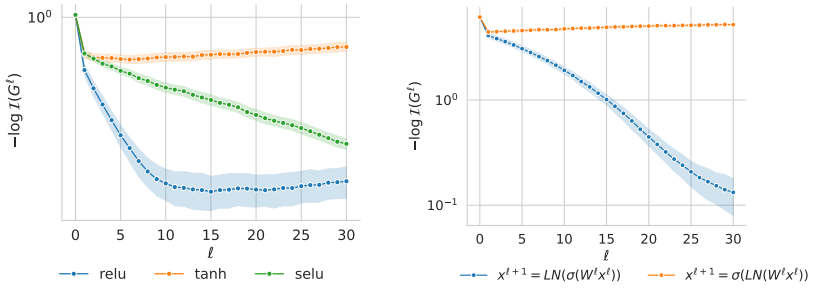


Figure 7: β_σ **correlates with SGD training speed.** Training loss (y-axis) vs epoch (x-axis) on a training dataset of `CIFAR10`. Each curve shows the MLP with corresponding activation, with color codes denoting isometry strength β_σ. Hyper parameters: depth: 10, width: 1000, batch size: 512, SGD with learning rate 0.01. Average of 5 independent runs.

7 Discussion

In this study, we explored the influence of layer normalization and nonlinear activation functions on the isometry of MLP representations. Our findings open up several avenues for future research.

Self normalized activations. It is worth investigating whether we can impose isometry without layer normalization. Our empirical observations suggest that certain activations, such as ReLU, require layer normalization to attain isometry. In contrast, other activations, which can be considered as "self-normalizing" (e.g., SeLU [Klambauer et al., 2017] and hyperbolic tangent), can achieve isometry with only offset and scale adjustments (see Figure 8). We experimentally show how we can replace centering and normalization by leveraging Hermit expansion of activation. Thus, we believe Hermit expansion provides a theoretical grounding to analyze the isometry of SeLU without centering and normalization.



(a) (b)

Figure 8: (a) Exploring the potential of achieving isometry with self-normalized activations. Batch size $n = 10$, width $d = 1000$. (b) Impact of the order of activation and normalization layers. Batch size $n = 10$, width $d = 1000$.

Impact of the ordering of normalization and activation layers on isometry. Theorem 4 highlights that the ordering of activation and normalization layers has a critical impact on the isometry. Figure 8 demonstrates that a different ordering can lead to a non-isotropic Gram matrix. Remarkably, the structure analyzed in this paper is used in transformers [Vaswani et al., 2017].

Normalization's role in stabilizing mean-field accuracy through depth. Numerous theoretical studies conjecture that mean-field predictions may not be reliable for considerably deep neural networks [Li et al., 2021, Joudaki et al., 2023]. Mean-field analysis encounters $O(1/\sqrt{\text{width}})$ error in each layer when the network width is finite. This error may accumulate with depth, making mean-field predictions increasingly inaccurate with an increase in depth. However, Figure 9 illustrates that layer normalization controls this error accumulation through depth. This might be attributable to the isometry bias induced by normalization, as proven in Theorem 1. Similarly, batch normalization also prevents error propagation with depth by imposing the same isometry [Joudaki et al., 2023]. This observation calls for future research on the essential role normalization plays in ensuring the accuracy of mean-field predictions.

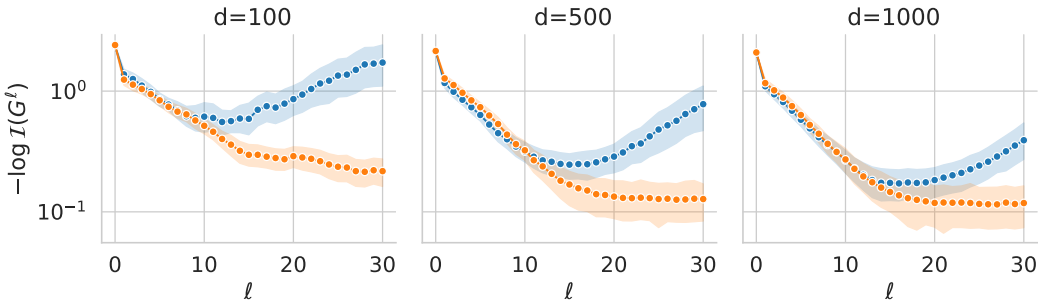


Figure 9: Role of normalization in stabilizing mean-field accuracy for ReLU, batch size $n = 10$. Mean-field normalization (blue): $x^{\ell+1} = \frac{a^\ell - c_0}{\sqrt{\bar{\sigma}(1)}}$. Layer normalization (orange): $x^{\ell+1} = \text{LN}(a^\ell - c_0)$

Acknowledgements

Amir Joudaki is funded through Swiss National Science Foundation Project Grant #200550 to Andre Kahles, and partially funded by ETH Core funding award to Gunnar Ratsch. We thank support from the NSF TRIPODS program (DMS-2022448).

References

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456. pmlr, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

Lorenzo Noci, Sotiris Anagnostidis, Luca Biggio, Antonio Orvieto, Sidak Pal Singh, and Aurelien Lucchi. Signal propagation in transformers: Theoretical perspectives and the role of rank collapse. *Advances in Neural Information Processing Systems*, 35:27198–27211, 2022a.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

Hadi Daneshmand, Jonas Kohler, Francis Bach, Thomas Hofmann, and Aurelien Lucchi. Batch normalization provably avoids ranks collapse for randomly initialised deep networks. *Advances in Neural Information Processing Systems*, 33:18387–18398, 2020.

Hadi Daneshmand, Amir Joudaki, and Francis Bach. Batch normalization orthogonalizes representations in deep random networks. *Advances in Neural Information Processing Systems*, 34: 4896–4906, 2021.

Hadi Daneshmand, Jason D Lee, and Chi Jin. Efficient displacement convex optimization with particle gradient descent. *International Conference on Machine Learning*, 2023.

Amir Joudaki, Hadi Daneshmand, and Francis Bach. On bridging the gap between mean field and finite width in deep random neural networks with batch normalization. *International Conference on Machine Learning*, 2023.

Jeffrey Pennington, Samuel Schoenholz, and Surya Ganguli. The emergence of spectral universality in deep networks. In *International Conference on Artificial Intelligence and Statistics*, pages 1924–1932, 2018.

Jonas Kohler, Hadi Daneshmand, Aurelien Lucchi, Thomas Hofmann, Ming Zhou, and Klaus Neymeyr. Exponential convergence rates for batch normalization: The power of length-direction decoupling in non-convex optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 806–815. PMLR, 2019.

Greg Yang, Jeffrey Pennington, Vinay Rao, Jascha Sohl-Dickstein, and Samuel S. Schoenholz. A mean field theory of batch normalization. In *International Conference on Learning Representations*, 2019.

Alexander G. de G. Matthews, Jiri Hron, Mark Rowland, Richard E. Turner, and Zoubin Ghahramani. Gaussian process behaviour in wide deep neural networks. In *International Conference on Learning Representations*, 2018.

Mufan Bill Li, Mihai Nica, and Daniel M. Roy. The neural covariance sde: Shaped infinite depth-and-width networks at initialization. *Advances in Neural Information Processing Systems*, 2022.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in Neural Information Processing Systems*, 31, 2018.

Nils Bjorck, Carla P. Gomes, Bart Selman, and Kilian Q. Weinberger. Understanding batch normalization. *Advances in Neural Information Processing Systems*, 31, 2018.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, pages 2793–2803, 2021.

Lorenzo Noci, Sotiris Anagnostidis, Luca Biggio, Antonio Orvieto, Sidak Pal Singh, and Aurelien Lucchi. Signal propagation in transformers: Theoretical perspectives and the role of rank collapse. *arXiv preprint arXiv:2206.03126*, 2022b.

Lechao Xiao, Yasaman Bahri, Jascha Sohl-Dickstein, Samuel Schoenholz, and Jeffrey Pennington. Dynamical isometry and a mean field theory of CNNs: How to train 10,000-layer vanilla convolutional neural networks. In *International Conference on Machine Learning*, pages 5393–5402, 2018.

Samuel S. Schoenholz, Justin Gilmer, Surya Ganguli, and Jascha Sohl-Dickstein. Deep information propagation. In *International Conference on Learning Representations*, 2017.

Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. *Advances in Neural Information Processing Systems*, 29, 2016.

Bobby He, James Martens, Guodong Zhang, Aleksandar Botev, Andrew Brock, Samuel L Smith, and Yee Whye Teh. Deep transformers without shortcuts: Modifying self-attention for faithful signal propagation. *arXiv preprint arXiv:2302.10322*, 2023.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Kunihiko Fukushima. Visual feature extraction by a multilayered network of analog threshold elements. *IEEE Transactions on Systems Science and Cybernetics*, 1969.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *Advances in Neural Information Processing Systems*, 30, 2017.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533, 2020.

Kaifeng Lyu, Zhiyuan Li, and Sanjeev Arora. Understanding the generalization benefit of normalization layers: Sharpness reduction. *Advances in Neural Information Processing Systems*, 35: 34689–34708, 2022.

Ekdeep S Lubana, Robert Dick, and Hidenori Tanaka. Beyond batchnorm: towards a unified understanding of normalization in deep learning. *Advances in Neural Information Processing Systems*, 34:4778–4791, 2021.

Amit Daniely, Roy Frostig, and Yoram Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. *Advances in Neural Information Processing Systems*, 29, 2016.

Greg Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *arXiv preprint arXiv:1902.04760*, 2019.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). URL `http://www.cs.toronto.edu/~kriz/cifar.html`.

Alexandru Meterez, Amir Joudaki, Francesco Orabona, Alexander Immer, Gunnar Rätsch, and Hadi Daneshmand. Towards training without depth limits: Batch normalization without gradient explosion. *arXiv preprint arXiv:2310.02012*, 2023.

Mufan Li, Mihai Nica, and Dan Roy. The future is log-gaussian: Resnets and their infinite-depth-and-width limit at initialization. *Advances in Neural Information Processing Systems*, 34:7852–7864, 2021.

Semyon Aranovich Gershgorin. Uber die abgrenzung der eigenwerte einer matrix. *News of the Russian Academy of Sciences. Mathematical series*, (6):749–754, 1931.

F Gustav Mehler. Ueber die entwicklung einer function von beliebig vielen variablen nach laplaceschen functionen höherer ordnung. *Journal für die Reine und Angewandte Mathematik (in German)*, 1866.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Appendix outline

The appendix is partitioned into four main components, each serving its purpose as described:

1. Section A demonstrates the proof of Theorem 1.
2. Section B details the proof of Theorem 4 alongside numerical confirmation of significant steps:
   - Section B.1 provides an elaborate review of the mean-field Gram dynamics.
   - Section B.2 presents the Lyapunov function γ, and establishes that this function provides both upper and lower bounds for the isometry, thereby implying that geometric contraction of $\gamma(G^\ell)$ indicates a geometric contraction of isometry gap $-\log \mathcal{I}(G^\ell)$.
   - Section B.3 proves that $\gamma(G^\ell)$ exhibits an exponential contraction in depth with rate β_σ.
3. Section C.1 explores the effect of gain on isometry, and links the rate to the associated isometry strength.
4. Section **??** outlines our rebuttal responses to the reviews that we chose to leave out of the main text.
   - Section C.3 explores the effect of varying widths of hidden layers on the isometry.
   - Section C.4 explores the notion of isometry for representations in language models.
5. Section D gives additional details concerning the experiments reported in the main text and appendix.

A Basic properties of isometry

Basic properties of isometry It is straightforward to check isometry obeys the following basic isometry-preserving properties:

Lemma A.1. *For PSD matrix M, the isometry defined in* (3) *obeys the following properties: 1) scale-invariance $\mathcal{I}(cM) = \mathcal{I}(M)$, 2) only takes value in the unit range $\mathcal{I}(M) \in [0, 1]$ 3) it takes its maximum value if and only if M is identity $\mathcal{I}(M) = 1 \iff M = I_n$, and 3) takes minimum value if and only if M is degenerate $\mathcal{I}(M) = 0$.*

Proof of Lemma A.1. The scale-invariance is trivially true as scaling M by any constant will scale $\det(M)^{1/n}$ and $\mathrm{tr}(M)$ by the same amount. The proof of other properties is a straightforward consequence of writing the isometry in terms of the eigenvalues $\mathcal{I}(M) = (\prod_i \lambda_i)^{1/n}/(\frac{1}{n}\sum_i \lambda_i)$, where λ_i's are eigenvalues of M. By arithmetic vs geometric mean inequality over the eigenvalues we have $(\prod_i \lambda_i)^{1/n} \leq \frac{1}{n}\sum_i \lambda_i)$, which proves that $\mathcal{I}(M) \in [0, 1]$. Furthermore, the inequality is tight iff the values are all equal $\lambda_1 = \cdots = \lambda_n$, which holds only for identity $M = I_n$. Finally, isometry is zero iff at least one eigenvalue is zero, which is the case for degenerate matrix M. □

B Proof of Theorem 4

B.1 Mean-field Gram Dynamics

Recall the mean-field Gram dynamics stated in equation (10):

$$G_*^{\ell+1} = \mathbb{E}_{h\sim N(0,G_*^\ell)}\left[\phi(\sigma(h))\phi(\sigma(h))^\top\right], \qquad \text{where } [\phi(a)]_i := (a_i - \mathbb{E}a_i)/\sqrt{\mathrm{Var}\, a_i}, \tag{12}$$

Assuming that inputs are encoded as columns of X, we can restate the MLP dynamics as follows

$$X^0 := \frac{1}{\sqrt{d}}\textsc{Ln}(X - \overline{X}), \qquad \text{inputs} \tag{13}$$

$$H^\ell := W^\ell X^\ell, W^\ell \sim N(0,1)^{d\times d}, \qquad \text{preactivation} \tag{14}$$

$$A^\ell := \sigma(H^\ell), \qquad \text{activations} \tag{15}$$

$$X^{\ell+1} = \frac{1}{\sqrt{d}}\textsc{Ln}(A^\ell - \overline{A^\ell}) \qquad \text{centering \& normalization,} \tag{16}$$

where centering and layer normalization are applied column-wise, as defined in the main text. Observe that Gram matrix of representations can be written as

$$G_d^{\ell+1} = {X^{\ell+1}}^\top X^{\ell+1} \tag{17}$$

$$= \frac{1}{d}\sum_{k=1}^d \left(\frac{A_{k1}^\ell - \mu_1}{s_1}, \ldots, \frac{A_{kn}^\ell - \mu_n}{s_n}\right)^{\otimes 2}, \quad \mu_i := \frac{1}{d}\sum_{k=1}^d (A_{ki}^\ell), s_i := \sqrt{\frac{1}{d}\sum_{k=1}^d (A_{ki}^\ell - \mu_i)^2}, \tag{18}$$

where $\otimes$ denotes Hadamard product, and subscript $_d$ emphasises the dependence of Gram on width d. Note that conditioned on the previous layer, rows of H^ℓ and A^ℓ are i.i.d. , because of independence of rows of W^ℓ. Thus, by law of large numbers, in the infinitely wide network regime, μ_i and s_i will converge to the expected mean and variance respectively $\lim_{d\to\infty} \mu_i = \mathbb{E}A^\ell_{1i}$, and $\lim_{d\to\infty} s_i = \sqrt{\text{Var}(A^\ell_{1i})}$, for all $i = 1, \ldots, n$. By construction of ϕ, in the infinitely wide regime, we can rewrite Gram dynamics as $\lim_{d\to\infty} G^\ell_d = \frac{1}{d}\sum_{k=1}^d \phi(A^\ell_{i\cdot})^\top \phi(A^\ell_{i\cdot})$. We can invoke the fact that rows of A^ℓ are i.i.d. to conclude that G_d is the sample Gram matrix that converges to its expectation

$$\lim_{d\to\infty} G^\ell_d = \mathbb{E}\phi(A^\ell_{1\cdot})\phi(A^\ell_{1\cdot})^\top = \mathbb{E}_{h\sim N(0,G^\ell)}\phi(\sigma(h))\phi(\sigma(h))^\top =: G^\ell_*, \tag{19}$$

where $_*$ denotes the mean-field regime $d \to \infty$. This concludes the connection between the mean-field Gram dynamics and infinitely wide Gram dynamics.

B.2 Introducing a potential

Here we will introduce a Lyapunov function that enables us to precisely quantify the isometry of activations in deep networks:

Definition 4. *Given a positive semidefinite matrix $G \in \mathbb{R}^{n\times n}$, we define $\gamma : \mathbb{R}^{n\times n} \to \mathbb{R}^{\geq 0}$ as:*

$$\gamma(G) := \max_{i\neq j} \frac{|\widetilde{G}_{ij}|}{1 - |\widetilde{G}_{ij}|}, \qquad \widetilde{G}_{ij} := G_{ij}/\sqrt{G_{ii}G_{jj}}. \tag{20}$$

Remarkably, γ obey exhibits an geometric contraction under one MLP layer update, which is stated in the following theorem:

Theorem B.1. *Let G be PSD matrix with unit diagonals $G_{ii} = 1, i = 1, \ldots, n$. It holds:*

$$\gamma\left(\mathbb{E}_{h\sim N(0,G)}\left[\phi(\sigma(h))\phi(\sigma(h))^\top\right]\right) \leq \gamma(G)/\beta_\sigma, \qquad \textit{where } [\phi(a)]_i := (a_i - \mathbb{E}a_i)/\sqrt{\text{Var}\, a_i}. \tag{21}$$

Thus, we may apply Theorem B.1 iteratively to prove that in the mean-field, the Lyapunov function $\gamma(G^\ell)$ decays at an exponential rate β_σ. A straightforward induction over layers leads to a decay rate in γ, which is presented in the next corollary.

Corollary B.2. *If activation σ has Hermite expansion with non-linearity strength β_σ, the mean-field Gram matrices G^ℓ_*, obey Lyapunov of these Gram matrices decays at an exponential rate β_σ:*

$$\gamma(G^\ell_*) \leq \gamma(G^0_*)\beta_\sigma^{-\ell}, \tag{22}$$

where G^0_* denotes the input Gram matrix.

In Figure B.1, we experimentally validated the above equation for MLPs with a finite width and activations {tanh ,relu, sigmoid} where we observe that the mean-field analysis well predicates the decay in γ.

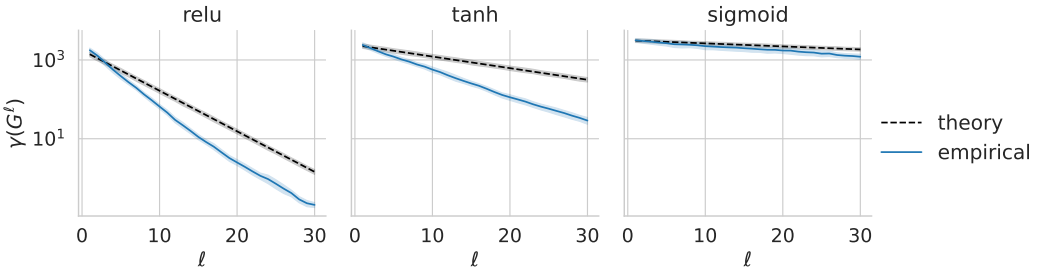


Figure B.1: $\gamma(G^\ell)$ vs depth ℓ, for MLP with $n = 10$, $d = 10000$, various activations σ. Solid lines shows average of 10 independent runs. The dashed traces the theoretical upper bounds given in Corollary B.2.

Interestingly, we can connect the Lyapunov γ to the isometry, by proving an upper and lower based on the determinant G based on $\gamma(G)$, when G is PSD and has unit diagonals:

Lemma B.3. *For PSD matrix G with unit diagonals holds:*

$$\left(1 - (n-1)\max_{i\neq j}|G_{ij}|\right)^n \leq \det(G) \leq 1 - \max_{i\neq j} G^2_{ij}, \tag{23}$$

where the lower bound holds if $(n-1)|G_{ij}| \leq 1$.

Now, we are ready to prove the main theorem.

Theorem B.4 (Restated Theorem 4)**.** *Let σ be an activation function with a Hermite expansion and a non-linearity strength β_σ, (see equation (9)). Given non-degenerate input Gram matrix G_*^0, then for sufficiently large layer $\ell \gtrsim \beta_\sigma^{-1}(-n \log \mathcal{I}(G_*^0) + \log(4n))$, we have*

$$-\log \mathcal{I}(G_*^\ell) \le \exp(-\ell \log \beta_\sigma - n \log \mathcal{I}(G_*^0) + \log(4n)). \tag{24}$$

Proof of Theorem 4. First, note that we have the following transformation

$$t = \frac{x}{1-x} \implies x = \frac{t}{t+1} \implies \max_{i \neq j} |G_{ij}| = \frac{\gamma(G)}{1+\gamma(G)}, \tag{25}$$

we have

$$\det(G_*^\ell) \ge (1 - (n-1) \max_{i \neq j} |G_{ij_*}^\ell|)^n \qquad \text{Lemma B.3} \tag{26}$$

$$\det(G_*^\ell) = \Big(1 - (n-1)(\gamma(G_*^\ell)/(1+\gamma(G_*^\ell)))\Big)^n \qquad \text{Invoke (25)} \tag{27}$$

$$\ge \Big(1 - (n-1)\gamma(G_*^\ell)\Big)^n \tag{28}$$

$$\implies -\frac{1}{n} \log \det(G_*^\ell) \le -\log(1 - (n-1)\gamma(G_*^\ell)) \tag{29}$$

$$\le 2n\gamma(G_*^\ell) \qquad \text{for } \gamma(G_*^\ell) < 1/(2n-2) \tag{30}$$

$$\le 2n\gamma(G_0)\beta_\sigma^{-\ell}. \tag{31}$$

By upper bound of Lemma B.3 we have $\max_{i \neq j} |G_{ij}| \le \sqrt{1 - \det(G)} \le 1 - \det(G)/2$, implying $\gamma(G_0) \le 2 \det(G_0)^{-1}$, which allows us to

$$-\log \mathcal{I}(G_*^\ell) \le 4n \det(G_0)^{-1} \beta_\sigma^{-\ell} \tag{32}$$

$$\le \exp(-\ell \log \beta_\sigma - n \log \mathcal{I}(G_0) + \log(4n)) \tag{33}$$

which concludes the proof. □

Proof of Lemma B.3. Lower bound. The lower bound is a result of Gershgorin circle theorem [Gershgorin, 1931], which implies that every eigenvalue must be within the disc $[1-(n-1)\max_{i\neq j}|G_{ij}|, 1+(n-1)\max_{i\neq j}|G_{ij}|]$. Thus, the determinant is lower-bounded by $(1-(n-1)\max_{i\neq j}|G_{ij}|)^n$.

Upper bound. Since G is PSD, we can write $G = X^\top X$, which implies that columns of X are unit norm $\|x_i\| = 1$, and G encodes the angles between them $\cos \angle(x_i, x_j) = \langle x_i, x_j \rangle = G_{ij}$. Furthermore, we have $\det(G) = \text{vol}(X)^2$, where the volume refers to the parallelepiped spanned by the columns of X. With this formulation, we write volume recursively as volume spanned by columns 1 up to $n-1$, times the projection distance of x_n from their span

$$\text{vol}(x_1, \ldots, x_n) = \text{dist}(x_n, \text{span}(x_1, \ldots, x_{n-1}))\text{vol}(x_1, \ldots, x_{n-1}),$$

where span refers to the space of all linear combinations of these vectors. Note that projection distance of x_i onto x_j can be written as $\sin \angle(x_i, x_j) = \sqrt{1 - \langle x_i, x_j \rangle^2} = \sqrt{1 - G_{ij}^2}$. Since the projection distance onto the linear span cannot be greater than projection distance onto a single vector, which is bounded by $\sqrt{1 - \max_{i \neq j} G_{ij}^2}$. This concludes the upper bound that $\det(G) \le 1 - \max_{i \neq j} G_{ij}^2$. □

B.3 Dual activation and proof of Thm. B.1

According to Daniely et al. [2016], we leverage the notion dual activation associated with the activation function σ, which is defined in following.

Definition 5. *Given activation $\sigma : \mathbb{R} \to \mathbb{R}$ that is square integrable with respect to the Gaussian kernel, define its dual activation $\hat{\sigma} : \mathbb{R} \to \mathbb{R}$, and its mean reduced dual $\bar{\sigma} : \mathbb{R} \to \mathbb{R}$ as:*

$$\hat{\sigma}(\rho) := \mathbb{E}\sigma(x)\sigma(y) \qquad \bar{\sigma}(\rho) := \mathbb{E}\left[(\sigma(x) - \mathbb{E}\sigma(x))(\sigma(y) - \mathbb{E}\sigma(y))\right] \tag{34}$$

The following lemma connects the dual activation and its mean-reduced version with the Hermite expansion of σ:

Lemma B.5. *Given two standard Gaussian variables $X, Y \sim N(0,1)$ with covariance $\mathbb{E}XY = \rho$, and activation σ with normalized Hermite coefficients c_{k_k}, we have*

$$\hat{\sigma}(\rho) = \sum_{k=0}^{\infty} c_k^2 \rho^k, \qquad \bar{\sigma}(\rho) = \sum_{k=1}^{\infty} c_k^2 \rho^k. \tag{35}$$

Finally, we have the tools to prove the first main theorem.

Proof of Theorem B.1. Let $a = \sigma(h)$ for $h \sim N(0, G)$. Note that by definition of the dual-activation we have $\mathbb{E}aa^\top = [\hat{\sigma}(G_{ij})]_{i,j\le n}$. Since we assumed G has unit diagonals, we have $h_i \sim N(0,1)$, which implies that $\mathbb{E}a_i = \mathbb{E}_{h_i\sim N(0,1)}\sigma(h_i) = c_0$, implying that $\mathbb{E}(a_i - \mathbb{E}a_i)(a_j - \mathbb{E}a_j)^\top = \hat{\sigma}(G_{ij}) - c_0^2 = \bar{\sigma}(G_{ij})$. Furthermore, the variance can be driven as $\mathrm{Var}(a_i) = \mathbb{E}a_i^2 - (\mathbb{E}a_i)^2 = \sum_{k=0}^\infty c_k^2 = \sum_{k=1}^\infty c_k^2 = \bar{\sigma}(1)$ for all $i = 1, \dots, n$. Thus, we have $\mathbb{E}\phi(a_i)\phi(a_j) = \bar{\sigma}(G_{ij})/\bar{\sigma}(1)$. In the matrix form we have

$$\gamma\left(\mathbb{E}_{h\sim N(0,G)}\left[\phi(\sigma(h))\phi(\sigma(h))^\top\right]\right) = \gamma\left([\bar{\sigma}(G_{ij})/\bar{\sigma}(1)]_{i,j\le n}\right) \tag{36}$$

The remainder proof relies on the following contractive property of Gram matrix potential:

Lemma B.6. *Consider activation σ, with normalized Hermite coefficients $\{c_k\}_{k\ge 0}$. For all $\rho \in (0,1)$, the mean-reduced dual activation $\bar{\sigma}$ obeys*

$$\frac{|\bar{\sigma}(\rho)|/\bar{\sigma}(1)}{1 - |\bar{\sigma}(\rho)|/\bar{\sigma}(1)} \le \beta_\sigma^{-1}\frac{|\rho|}{1-|\rho|}, \tag{37}$$

which the right hand-side is strictly larger if some nonlinear coefficient is nonzero $c_k \neq 0$ for some $k \ge 2$.

Thus we can apply Lemma B.6 on each element $i \neq j$ to conclude that

$$\begin{aligned}
\frac{|\bar{\sigma}(G_{ij})|/\bar{\sigma}(1)}{1 - |\bar{\sigma}(G_{ij})|/\bar{\sigma}(1)} &\le \frac{|G_{ij}|}{1-|G_{ij}|}\beta_\sigma^{-1} && \text{Lemma B.6 for all } i \neq j && (38)\\
&\le \beta_\sigma^{-1}\max_{i\neq j}\frac{|G_{ij}|}{1-|G_{ij}|} && && (39)\\
&= \beta_\sigma^{-1}\gamma(G). && && (40)\\
& && && (41)
\end{aligned}$$

since the inequality holds for any value of $i \neq j$, we can take the maximum over $i \neq jj$ to write:

$$\gamma\left(\bar{\sigma}(G)/\bar{\sigma}(1)\right) = \max_{i\neq j}\frac{|\bar{\sigma}(G_{ij})|/\bar{\sigma}(1)}{1 - |\bar{\sigma}(G_{ij})|/\bar{\sigma}(1)} \le \beta_\sigma^{-1}\gamma(G), \tag{42}$$

which concludes the proof. □

Proof of Lemma B.6. Note the ratio is invariant to scaling of $\sum_{k=1}^\infty c_k^2$. Hence, we assume $\sum_{k=1}^\infty c_k^2 = 1$ without loss of generality. With this simplification, we have $\bar{\sigma}(1) = 1$. For the positive range $\rho \in [0,1]$ we have

$$\begin{aligned}
(\frac{\bar{\sigma}(\rho)}{1-\bar{\sigma}(\rho)})(\frac{\rho}{1-\rho})^{-1} &= \frac{\rho^{-1}\sum_{k=1}^\infty c_k^2\rho^k}{(1-\rho)^{-1}\sum_{k=1}^\infty c_k^2(1-\rho^k)} && (43)\\
&= \frac{\sum_{k=1}^\infty c_k^2\rho^{k-1}}{\sum_{k=1}^\infty c_k^2(\frac{1-\rho^k}{1-\rho})} && (44)\\
&= \frac{\sum_{k=1}^\infty c_k^2\rho^{k-1}}{\sum_{k=1}^\infty c_k^2\left(\sum_{i=0}^{k-1}\rho^i\right)} && (45)\\
&\le \frac{\sum_{k=1}^\infty c_k^2\rho^{k-1}}{(\sum_{k=1}^\infty c_k^2\rho^{k-1}) + \sum_{k=2}^\infty c_k^2} && (46)\\
&\le \max_{\rho\in[0,1]}\frac{\sum_{k=1}^\infty c_k^2\rho^{k-1}}{(\sum_{k=1}^\infty c_k^2\rho^{k-1}) + \sum_{k=2}^\infty c_k^2} && (47)\\
&= \frac{\sum_{k=1}^\infty c_k^2}{\sum_{k=1}^\infty c_k^2 + \sum_{k=1}^\infty c_k^2 - c_1^2} && (48)\\
&= \frac{1}{\beta_\sigma}. && (49)
\end{aligned}$$

Thus for $\rho \in [0,1]$ we have

$$\frac{\bar{\sigma}(\rho)}{1-\bar{\sigma}(\rho)} \le \frac{\rho}{1-\rho}\beta_\sigma^{-1}. \tag{50}$$

By Jensen inequality for convex function $x \mapsto |x|$ we have

$$|\bar{\sigma}(\rho)| = |\sum_{k=1}^\infty c_k^2\rho^k| \le \sum_{k=1}^\infty c_k^2|\rho|^k = \bar{\sigma}(|\rho|) \tag{51}$$

Because $x \mapsto x/(1-x)$ is monotonically increasing for $x \in [0,1]$, we have

$$\frac{|\bar{\sigma}(\rho)|}{1-|\bar{\sigma}(\rho)|} \leq \frac{\bar{\sigma}(|\rho|)}{1-\bar{\sigma}(|\rho|)} \leq \frac{|\rho|}{1-|\rho|}\beta_\sigma^{-1}. \tag{52}$$

Where we invoked the inequality that was proven for $\rho \in [0,1]$, because $|\rho| \in [0,1]$. □

The following lemma, which is a consequence of Mehler's formula, is at the hart of proof of Lemma B.5:

Lemma B.7 (Consequence of Mehler's kernel)**.** *If $X, Y \sim N(0,1)$ with covariance $EXY = \rho$ we have*

$$E_{X,Y} He_n(X) He_k(Y) = \rho^n \delta_{nk}$$

where δ_{nk} is the Dirac delta.

Proof of Lemma B.5. Let $\sigma(x) = \sum_{k=0} c_k He_k(x)$, denote the Hermite expansion of σ. Thus, we have

$$\hat{\sigma}(\rho) := \mathbb{E}_{X,Y}\sigma(X)\sigma(Y) \tag{53}$$

$$= \sum_{n,k=0}^{\infty} c_n c_k \mathbb{E}_{X,Y} He_n(X) He_k(Y) \tag{54}$$

$$= \sum_{k=0}^{\infty} c_k^2 \rho^k, \tag{55}$$

where in the last line we applied result of Lemma B.7. For the mean-reduced dual kernel $\bar{\sigma}$, observe that $\mathbb{E}_{X\sim N(0,1)}\sigma(X) = c_0$. Thus, the reduction of mean will cancell the $k=0$ term, which concludes the proof that $\bar{\sigma}(\rho) = \sum_{k=1} c_k^2 \rho^k$. □

Proof of Lemma B.7. The property can be deduced from Mehler's formula Mehler [1866]. The formula states that

$$\frac{1}{\sqrt{1-\rho^2}} \exp\left(-\frac{\rho^2(x^2+y^2) - 2xy\rho}{2(1-\rho^2)}\right) \tag{56}$$

$$= \sum_{m=0}^{\infty} He_m(x) He_m(y), \tag{57}$$

where the $m!$ factor difference is due to the definition of Hermite polynomials with an additional $1/\sqrt{m!}$ compared to the one used in Mehler's kernel. Observe that the left hand side is equal to $p(x,y)/p(x)p(y)$, where $p(x,y)$ is the joint PDF of (X,Y), and $p(x), p(y)$ are PDF of X and Y respectively. Therefore, we can take the expectation using the expansion

$$\mathbb{E}_{X,Y}\left[He_n(X)He_k(Y)\right] = \int He_n(x)He_k(y)p(x,y)dxdy \tag{58}$$

$$= \sum_{m=0}^{\infty} \rho^m \int He_n(x)He_k(y)He_m(x)He_m(y)dp(x)dp(y) \tag{59}$$

$$= \sum_{m=0}^{\infty} \rho^m \mathbb{E}_{X\sim N(0,1)}\left[He_n(X)He_m(X)\right] \mathbb{E}_{Y\sim N(0,1)}\left[He_k(Y)He_m(Y)\right] \tag{60}$$

$$= \rho^n \delta_{nk} \tag{61}$$

where in the last line we used the orthogonality property $E_{X\sim N(0,1)} H_k(x) H_n(X) = \delta_{nk}$. □

C Additional experiments

C.1 Quantifying the influence of gain on isometry through non-linearity strength

The concept of gain in neural networks is vital and closely connected with the weights initialization. A neural network with properly initialized weights can learn faster, have a lesser chance of getting stuck at sub-optimal solutions, and provide better generalization. The impact of gain can be visualized through the lens of weight initialization strategies such as Xavier normalization [Glorot and Bengio, 2010], which has shown significant effectiveness in optimizing neural networks. These initialization strategies apply a gain value to the weights, which is a scaling factor, to ensure a good signal flow through many layers during the forward and backward passes. The gain value essentially determines the variance of the weights in the initialization stage.

σ	$\exp(\alpha x)$	$\sin(\alpha x)$	$\max(\alpha x, 0)$
β_σ	$\frac{2-2e^{\alpha^2}+\alpha^2}{1-e^{\alpha^2}}$	$\frac{2(-1+e^{2\alpha^2}-e^{\alpha^2}\alpha^2)}{-1+e^{2\alpha^2}}$	$\frac{4-3\pi}{2-2\pi}$

Table C.1: Relationship between non-linearity strength and gain.

As an extension to our prior investigations, we delve into understanding the influence of gain on isometry, predominantly through our calculated metric, the non-linearity strength, denoted as β_σ, as a function of gain α. For certain instances, such as ReLU, sine, and exponential activations, we are capable of deriving $\beta_\sigma(\alpha)$ in a closed form. Table C.1 presents a few of these cases.

For a more extensive selection of activation functions, we have numerically computed the non-linearity strength as a function of gain α, as visualized in Figure C.1. Leveraging Theorem 4, these computed values provide an estimation for the isometry strength for various activations. This correlation proves to be remarkably predictive, as shown in Figure C.2. Remarkably, in the case of ReLU activation, its unique characteristics lead to both our closed form β_σ (refer Table C.1) and rate towards isometry (refer Figure C.2) remaining consistent across different values of α.

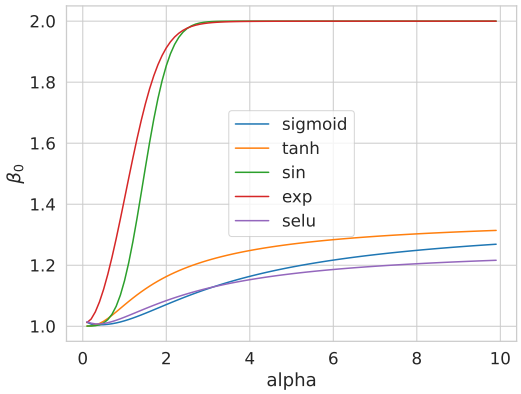


Figure C.1: Analysis of gain through the lens of isometry strength.

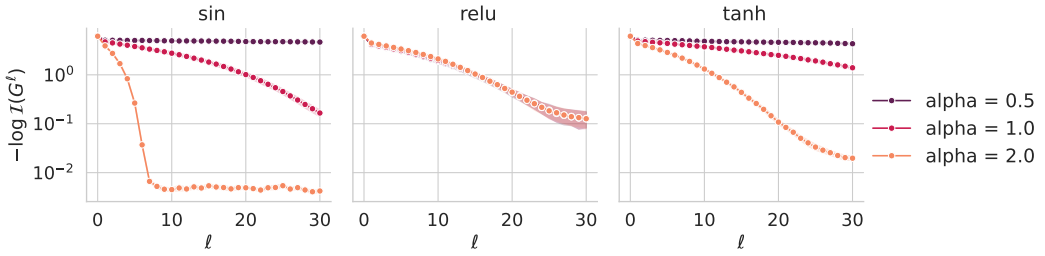


Figure C.2: Influence of gain on the attainment of isometry.

C.2 Comparison to Xavier gain for initialization

Inspired by the results so far, we can compare mean-field centering and normalization to Xavier gain for activations. Figure C.3 demonstrates that all mean-field based gains improve the isometry when compared with Xavier initialization. However, this is markedly stronger for ReLU and leaky ReLU. We can explain this starker contrast by the fact that both activations have a significant offset term c_0, which is not corrected by the Xavier initialization.

C.3 Varying width of hidden layers

While in our theoretical setup, we assume the network width is constant across the layers, this is only a choice to streamline our proof and notation. Since our primary result is derived from the mean-field regime, the only criterion for it to hold is for the width to be sufficiently large to approximate the mean-field regime. Our

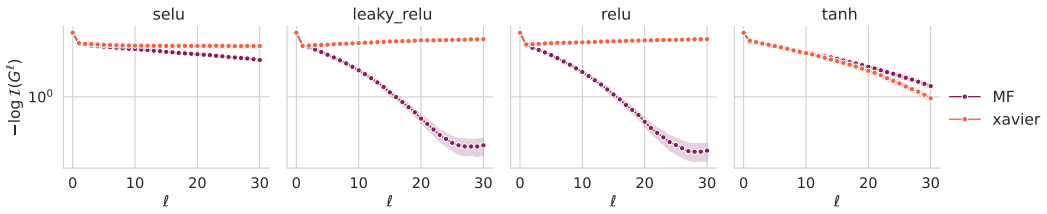


Figure C.3: Isometry vs depth for mean-field centering and normalization to Xavier initialization.

experiments in Figure C.4 substantiate this claim that the specific sizes of hidden layers, as long as they are large, will not impact our main results on the isometry. We empirically validate this for four different configurations and show that the decay of the isometry gap remains largely consistent across these configurations.

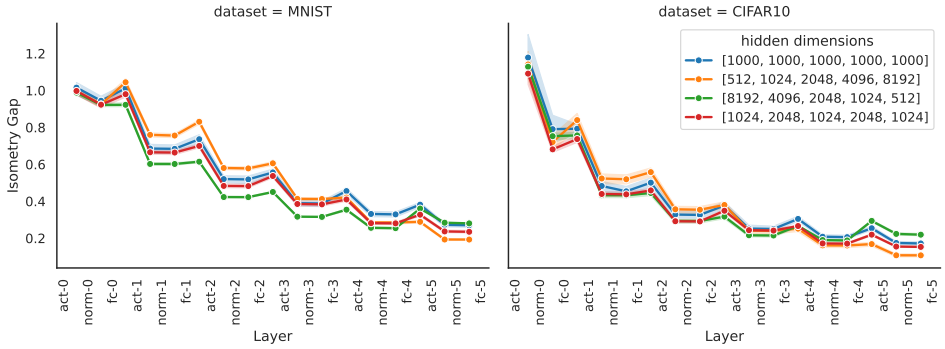


Figure C.4: **Theorem 4 holds for variable width MLP and MNIST**. Isometry gap of various layers for MLP with ReLU activation and fixed width (blue), growing width (orange), narrowing width (green), and interchanging width (red) on MNIST (left) and CIFAR10 (right) training data. The weights of MLP are random. These plots show that our central theory holds regardless of the dataset or shape of MLP widths.

C.4 Isometry in pretrained large language models

Since our theory for normalization is not limited to initialization, we can expand our search for isometry to other architectures. Figure C.5 shows the important role of normalization in the pre-trained GPT2 network. However, we need to adjust the notion of isometry with the architecture of layer norm in a transformer in mind. In fact, the mean and standard deviation are computed over features separately for each token. Thus, to adapt the notion of isometry, we can view each token as a sample and define Gram over different tokens. Thus, isometry here quantifies the similarity between various tokens within one sample. As can be seen in the figure below, LayerNorm layers (shaded in red) in the last six layers of the pre-trained GPT2 increase the isometry between tokens, which is consistent with our theory of layer normalization. It is crucial that our theory holds deterministically, which extends to the pre-trained model.

One caveat in interpreting our results is that, in practice, LayerNorm layers have learnable parameters that make them deviate from our theory. It would be fruitful to study the effects of learned parameters to discern it from the role of centering and normalization for a future study.

D Details about empirical validations

Hardware. All experiments, except for Figure 7 were run on a single `AMD Ryzen 9 3900X 12-Core` CPU, which takes about 5 minutes overall. For Figure 7, we trained our models on a single `NVIDIA GeForce RTX 3090` GPU, which takes about 3 minutes.

Figures. The solid lines in all plots represent the average performance over multiple independent runs, and the shaded regions indicate the confidence intervals. Unless stated otherwise, each average is computed over #10 independent runs.

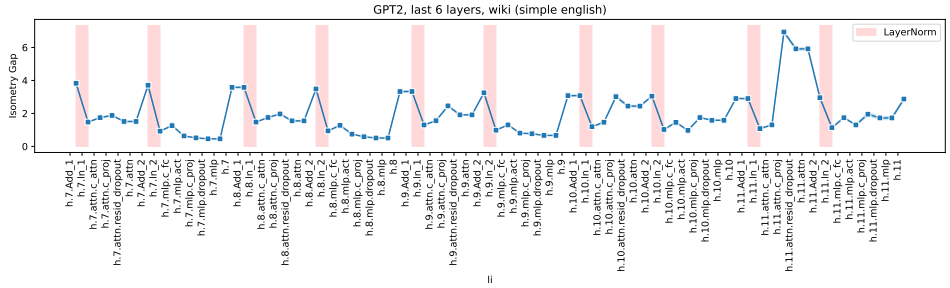


Figure C.5: **Validation of corollary 2 in GPT2**. The last six transformer layers of pre-trained GPT2, LayerNorm layers (shaded; red) decrease the isometry gap.

Codes and reproducibility. We implemented our experiments in Python using the PyTorch framework Paszke et al. [2019]. All the figures are reproducible with the code attached in the supplementary.

Training procedure for Figure 7 In each epoch, the isometry gap are computed per each batch, and then averaged over the entire test set. The epoch i corresponds to the network at after i steps of training on the training set of CIFAR10 (epoch 0 means network is at initialization).