Dynamical Isometry and a Mean Field Theory of CNNs: How to Train 10,000-Layer Vanilla Convolutional Neural Networks

Lechao Xiao [1 2] **Yasaman Bahri** [1 2] **Jascha Sohl-Dickstein** [1] **Samuel S. Schoenholz** [1] **Jeffrey Pennington** [1]

Abstract

In recent years, state-of-the-art methods in computer vision have utilized increasingly deep convolutional neural network architectures (CNNs), with some of the most successful models employing hundreds or even thousands of layers. A variety of pathologies such as vanishing/exploding gradients make training such deep networks challenging. While residual connections and batch normalization do enable training at these depths, it has remained unclear whether such specialized architecture designs are truly necessary to train deep CNNs. In this work, we demonstrate that it is possible to train vanilla CNNs with ten thousand layers or more simply by using an appropriate initialization scheme. We derive this initialization scheme theoretically by developing a mean field theory for signal propagation and by characterizing the conditions for *dynamical isometry*, the equilibration of singular values of the input-output Jacobian matrix. These conditions require that the convolution operator be an orthogonal transformation in the sense that it is norm-preserving. We present an algorithm for generating such random initial orthogonal convolution kernels and demonstrate empirically that they enable efficient training of extremely deep architectures.

1. Introduction

Deep convolutional neural networks (CNNs) have been crucial to the success of deep learning. Architectures based on CNNs have achieved unprecedented accuracy in domains ranging across computer vision (Krizhevsky et al., 2012), speech recognition (Hinton et al., 2012), natural language processing (Collobert et al., 2011; Kalchbrenner et al., 2014; Kim, 2014), and recently even the board game Go (Silver et al., 2016; 2017).

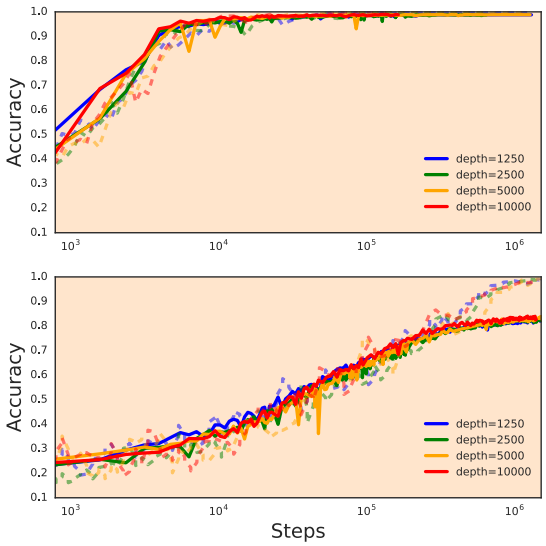

Figure 1. Extremely deep CNNs can be trained without the use of batch normalization or residual connections simply by using a Delta-Orthogonal initialization with critical weight and bias variance and appropriate (in this case, $\tanh$) nonlinearity. Test (solid) and training (dashed) curves on MNIST (top) and CIFAR-10 (bottom) for depths $1{,}250$, $2{,}500$, $5{,}000$, and $10,000$.

The performance of deep convolutional networks has improved as these networks have been made ever deeper. For example, some of the best-performing models on ImageNet (Deng et al., 2009) have employed hundreds or even a thousand layers (He et al., 2016a;b). However, these extremely deep architectures have been trainable only in conjunction with techniques like residual connections (He et al., 2016a) and batch normalization (Ioffe & Szegedy, 2015). It is an open question whether these techniques qualitatively improve model performance or whether they are necessary crutches that solely make the networks easier to train. In this work, we study vanilla CNNs using a combination of theory and experiment to disentangle the notions of trainability and generalization performance. In doing so, we show that through a careful, theoretically-motivated initialization scheme, we can train vanilla CNNs with 10,000 layers using no architectural tricks.

[1]Google Brain [2]Work done as part of the Google AI Residency program (g.co/airesidency). Correspondence to: Lechao Xiao <xlc@google.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

Recent work has used mean field theory to build a theoretical understanding of neural networks with random parameters (Poole et al., 2016; Schoenholz et al., 2017; Yang & Schoenholz, 2017; Schoenholz et al., 2017; Karakida et al., 2018; Hayou et al., 2018; Hanin & Rolnick, 2018; Yang & Schoenholz, 2018). These studies revealed a maximum depth through which signals can propagate at initialization, and verified empirically that networks are trainable precisely when signals can travel all the way through them. In the fully-connected setting, the theory additionally predicts the existence of an order-to-chaos phase transition in the space of initialization hyperparameters. For networks initialized on the critical line separating these phases, signals can propagate indefinitely and arbitrarily deep networks can be trained. While mean field theory captures the "average" dynamics of random neural networks it does not quantify the scale of gradient fluctuations that are crucial to the stability of gradient descent. A related body of work (Saxe et al., 2013; Pennington et al., 2017; 2018) has examined the input-output Jacobian and used random matrix theory to quantify the distribution of its singular values in terms of the activation function and the distribution from which the initial random weight matrices are drawn. These works concluded that networks can be trained most efficiently when the Jacobian is well-conditioned, a criterion that can be achieved with orthogonal, but not Gaussian, weight matrices. Together, these approaches have allowed researchers to efficiently train extremely deep network architectures, but so far they have been limited to neural networks composed of fully-connected layers.

In the present work, we continue this line of research and extend it to the convolutional setting. We show that a well-defined mean-field theory exists for convolutional networks in the limit that the number of channels is large, even when the size of the image is small. Moreover, convolutional networks have precisely the same order-to-chaos transition as fully-connected networks, with vanishing gradients in the ordered phase and exploding gradients in the chaotic phase. And just like fully-connected networks, very deep CNNs that are initialized on the critical line separating those two phases can be trained with relative ease.

Moving beyond mean field theory, we additionally show that the random matrix analysis of (Pennington et al., 2017; 2018) carries over to the convolutional setting. Furthermore, we identify an efficient construction from the wavelet literature that generates random orthogonal matrices with the block-circulant structure that corresponds to convolution operators. This construction facilitates random orthogonal initialization for convolulational layers and enables good conditioning of the end-to-end Jacobian matrices of arbitrarily deep networks. We show empirically that networks with this initialization can train significantly more quickly than standard convolutional networks.

Finally, we emphasize that although the order-to-chaos phase boundaries of fully-connected and convolutional networks look identical, the underlying mean-field theories are in fact quite different. In particular, a novel aspect of the convolutional theory is the existence of multiple depth scales that control signal propagation at different spatial frequencies. In the large depth limit, signals can only propagate along modes with minimal spatial structure; all other modes end up deteriorating, even at criticality. We hypothesize that this type of signal degradation is harmful for generalization, and we develop a modified initialization scheme that allows for balanced propagation of signals among all frequencies. In this scheme, which we call Delta-Orthogonal initialization, the orthogonal kernel is drawn from a spatially non-uniform distribution, and it allows us to train vanilla CNNs of 10,000 layers or more with no degradation in performance.

2. Theoretical results

In this section, we first derive a mean field theory for signal propagation in random convolutional neural networks. We will follow the general methodology established in Poole et al. (2016); Schoenholz et al. (2017); Yang & Schoenholz (2017). We will then arrive at a theory for the singular value distribution of the Jacobian following Pennington et al. (2017; 2018). Together, this will allow us to derive theoretically motivated initialization schemes for convolutional neural networks that we call orthogonal kernels and Delta-Orthogonal kernels. Later we will demonstrate experimentally that these kernels outperform existing initialization schemes for very deep vanilla convolutional networks.

2.1. A mean field theory for CNNs

2.1.1. Recursion relation for covariance

Consider an L-layer 1D[1] CNN with periodic boundary conditions, filter width $2k+1$, number of channels c, spatial size n, per-layer weight tensors $\omega^l \in \mathbb{R}^{(2k+1)\times c\times c}$, and biases $b^l \in \mathbb{R}^c$. Let $\phi : \mathbb{R} \to \mathbb{R}$ be the activation function and let $h^l_j(\alpha)$ denote the pre-activation unit at layer l, channel j, and spatial location $\alpha \in sp$, where we define the set of spatial locations $sp = \{1, ..., n\}$. The forward-propagation dynamics can be described by the recurrence relation,

$$h^{l+1}_j(\alpha) = \sum_{\substack{i\in chn\\ \beta\in ker}} \phi(h^l_i(\alpha+\beta))\omega^{l+1}_{ij}(\beta) + b^{l+1}_j, \quad (2.1)$$

where $ker = \{\beta \in \mathbb{Z} : |\beta| \le k\}$ and $chn = \{1, \dots, c\}$. At initialization, we take the weights $\omega^l_{ij}(\beta)$ to be drawn i.i.d. from the Gaussian $\mathcal{N}(0, \sigma^2_\omega/(c(2k+1)))$ and the biases b^l_j

[1] For notational simplicity, we consider one-dimensional convolutions, but the d-dimensional case proceeds identically.

to be drawn i.i.d. from the Gaussian $\mathcal{N}(0, \sigma_b^2)$. Note that $h_i^l(\alpha) = h_i^l(\alpha + n) = h_i^l(\alpha - n)$ since we assume periodic boundary conditions. We wish to understand how signals propagate through these networks. As in previous work in this vein, we will take the large network limit, which in this context corresponds to the number of channels $c \to \infty$. This allows us to use powerful theoretical tools such as mean field theory and random matrix theory. Moreover, this approximation has been shown to give results that agree well with experiments on finite-size networks.

In the limit of a large number of channels, the central limit theorem implies that the pre-activation vectors h_j^l are i.i.d. Gaussian with mean zero and covariance matrix $\Sigma^l_{\alpha,\alpha'} = \mathbb{E}[h_j^l(\alpha) h_j^l(\alpha')]$. Here, the expectation is taken over the weights and biases and it is independent of the channel index j. In this limit, the covariance matrix takes the form (see Supplemental Materials (SM)),

$$\Sigma^{l+1}_{\alpha,\alpha'} = \sigma_b^2 + \frac{\sigma_w^2}{2k+1} \sum_{\beta \in ker} \mathbb{E}\big[\phi(h_j^l(\alpha+\beta))\phi(h_j^l(\alpha'+\beta))\big] \,, \tag{2.2}$$

and is independent of j. A more compact representation of this equation can be given as,

$$\boldsymbol{\Sigma}^{l+1} \equiv \mathcal{A} \star \mathcal{C}(\boldsymbol{\Sigma}^l) \,, \tag{2.3}$$

where $\mathcal{A} = \frac{1}{2k+1}\mathbf{I}_{2k+1}$ and $\star$ denotes 2D circular cross-correlation, i.e. for any matrix $\boldsymbol{C}$, $\mathcal{A} \star \boldsymbol{C}$ is defined as,

$$[\mathcal{A} \star \mathbf{C}]_{\alpha,\alpha'} = \frac{1}{2k+1} \sum_{\beta \in ker} C_{\alpha+\beta,\alpha'+\beta} \,. \tag{2.4}$$

The function $\mathcal{C} : \mathrm{PSD}_n \to \mathrm{PSD}_n$ is related to the $\mathcal{C}$-map defined in Poole et al. (2016) (see also (Daniely et al., 2016)) and is given by,

$$[\mathcal{C}(\boldsymbol{\Sigma})]_{\alpha,\alpha'} = \sigma_\omega^2 \, \mathbb{E}_{\boldsymbol{h} \sim \mathcal{N}(0,\boldsymbol{\Sigma})} \left[\phi(h_\alpha)\phi(h_{\alpha'})\right] + \sigma_b^2 . \tag{2.5}$$

All but the two dimensions α and α' in eqn. (2.5) marginalize, so, as in (Poole et al., 2016), the $\mathcal{C}$-map can be computed by a two-dimensional integral. Unlike in (Poole et al., 2016), α and α' do not correspond to different examples but rather to different spatial positions and eqn. (2.5) characterizes how signals from a single input propagate through convolutional networks in the mean-field approximation[2].

2.1.2. Dynamics of signal propagation

We now seek to study the dynamics induced by eqn. (2.3). Schematically, our approach will be to identify fixed points of eqn. (2.3) and then linearize the dynamics around these fixed points. These linearized dynamics will dictate the stability and rate of decay towards the fixed points, which determines the depth scales over which signals in the network can propagate.

Schoenholz et al. (2017) found that for many activation functions ϕ (e.g. $\tanh$) and any choice of σ_w and σ_b, the $\mathcal{C}$-map has a fixed point $\boldsymbol{\Sigma}^*$ (i.e. $\mathcal{C}(\boldsymbol{\Sigma}^*) = \boldsymbol{\Sigma}^*$) of the form,

$$\Sigma^*_{\alpha,\alpha'} = q^*(\delta_{\alpha,\alpha'} + (1 - \delta_{\alpha,\alpha'})c^*) \,, \tag{2.6}$$

where $\delta_{a,b}$ is the Kronecker-δ, q^* is the fixed-point variance of a single input, and c^* is the fixed-point correlation between two inputs. It follows from the form of eqn. (2.4) that $\boldsymbol{\Sigma}^*$ is also a fixed point of the layer-to-layer covariance map in the convolutional case (eqn. (2.3)), i.e. $\boldsymbol{\Sigma}^* = \mathcal{A} \star \mathcal{C}(\boldsymbol{\Sigma}^*)$.

To analyze the dynamics of the iteration map (2.3) near the fixed point $\boldsymbol{\Sigma}^*$, we define $\boldsymbol{\epsilon}^l = \boldsymbol{\Sigma}^* - \boldsymbol{\Sigma}^l$ and expand eqn. (2.3) to lowest order in $\boldsymbol{\epsilon}$. This expansion requires the Jacobian of the $\mathcal{C}$-map evaluated at the fixed point, the properties of which we analyze in the SM. In brief, perturbations in q^* and c^* evolve independently and the Jacobian decomposes into a diagonal eigenspace V_d with eigenvalue χ_{q^*}, and an off-diagonal eigenspace $V_\mathrm{o.d.}$ with eigenvalue χ_{c^*}. The eigenvalues are given by[3],

$$\begin{aligned} \chi_{c^*} &= \sigma_w^2 \mathbb{E}_{\boldsymbol{h} \sim \mathcal{N}(0,\boldsymbol{C}^*)}[\phi'(h_1)\phi'(h_2)] \,, \; h_1 \neq h_2 \,, \\ \chi_{q^*} &= \sigma_w^2 \mathbb{E}_{\boldsymbol{h} \sim \mathcal{N}(0,\boldsymbol{C}^*)}[\phi''(h_1)\phi(h_1) + \phi'(h_1)^2] \,, \end{aligned} \tag{2.7}$$

and the eigenspaces have bases,

$$\begin{aligned} B_\mathrm{d} &= \{M^{\alpha,\alpha} : M^{\alpha,\alpha}_{\bar\alpha,\bar\alpha'} = \gamma\delta_{\alpha,\bar\alpha}\delta_{\alpha,\bar\alpha'} + \delta_{\bar\alpha,\alpha} + \delta_{\bar\alpha',\alpha}\}_{\alpha \in sp} \\ B_\mathrm{o.d.} &= \{M^{\alpha,\alpha'} : M^{\alpha,\alpha'}_{\bar\alpha,\bar\alpha'} = \delta_{\alpha,\bar\alpha}\delta_{\alpha',\bar\alpha'} + \delta_{\alpha,\bar\alpha'}\delta_{\alpha',\bar\alpha}\}_{\alpha \neq \alpha'} \,, \end{aligned} \tag{2.8}$$

i.e. $V_\mathrm{d} = \mathrm{span}(B_\mathrm{d})$ and $V_\mathrm{o.d} = \mathrm{span}(B_\mathrm{o.d.})$. Note that χ_{q^*} and χ_{c^*} also were found in Schoenholz et al. (2017) to control signal propagation in the fully-connected case. The constant γ is given in Lemma B.2 of the SM but does not concern us here. This eigen-decomposition implies that the layer-wise deviations from the fixed point evolve under eqn. (2.3) as,

$$\boldsymbol{\epsilon}^{l+1} = \chi_{q^*} \mathcal{A} \star \boldsymbol{\epsilon}^l_\mathrm{d} + \chi_{c^*} \mathcal{A} \star \boldsymbol{\epsilon}^l_\mathrm{o.d.} + \mathcal{O}((\boldsymbol{\epsilon}^l)^2) \,, \tag{2.9}$$

where $\boldsymbol{\epsilon}_\mathrm{d}$ and $\boldsymbol{\epsilon}_\mathrm{o.d.}$ are decomposition of $\boldsymbol{\epsilon}$ into the eigenspaces V_d and $V_\mathrm{o.d.}$.

Eqn. (2.9) defines the linear dynamics of random convolutional neural networks near their fixed points and is the basis for the in-depth analysis of the following subsections.

[2] The multi-input analysis proceeds in precisely the same manner as we present here, but comes with increased notational complexity and features no qualitatively different behavior, so we focus our presentation on the single-input case.

[3] By the symmetry of $\boldsymbol{\Sigma}^*$, these expectations are independent of spatial location and of the choice of h_1 and h_2.

2.1.3. Multi-dimensional signal propagation

In the fully-connected setting, the dynamics of signal propagation near the fixed point are governed by scalar evolution equations. In contrast, the convolutional setting enjoys much richer dynamics, as eqn. (2.9) describes a multi-dimensional system that we now analyze.

It follows from eqns. (2.4) and (2.8) (see also the SM) that $\mathcal{A}$ does not mix the diagonal and off-diagonal eigenspaces, i.e. $\mathcal{A} \star \boldsymbol{\epsilon}_{\text{d}} \in V_{\text{d}}$ and $\mathcal{A} \star \boldsymbol{\epsilon}_{\text{o.d.}} \in V_{\text{o.d.}}$. To see this, note that for $M^{\alpha,\alpha'} \in V_{\text{o.d.}}$, the definition implies $M^{\alpha,\alpha'}_{\bar{\alpha}+\beta,\bar{\alpha}'+\beta} = M^{\alpha-\beta,\alpha'-\beta}_{\bar{\alpha},\bar{\alpha}'}$. This property ensures that $\mathcal{A} \star M^{\alpha,\alpha'}$ can be expressed as a linear combination of matrices in $V_{\text{o.d.}}$, which means it also belongs to $V_{\text{o.d}}$. The same argument applies to $M^{\alpha,\alpha} \in V_{\text{d.}}$. As a result, these eigenspaces evolve entirely independently under the linearization of the covariance iteration map (2.3).

Let l_0 denote the depth over which transient effects persist and after which eqn. (2.9) accurately describes the linearized dynamics. Therefore, at depths larger than l_0, we have

$$\boldsymbol{\epsilon}^l \approx \underbrace{\mathcal{A} \star \cdots \mathcal{A} \star}_{l-l_0} (\chi_{q^*}^{l-l_0} \boldsymbol{\epsilon}_{\text{d}}^{l_0} + \chi_{c^*}^{l-l_0} \boldsymbol{\epsilon}_{\text{o.d.}}^{l_0}) \,. \tag{2.10}$$

This matrix-valued equation is still somewhat complicated owing to the nested applications of $\mathcal{A}$. To further elucidate the dynamics, we can move to a Fourier basis, which diagonalizes the circular cross-correlation operator and decouples the modes of eqn. (2.10). In particular, let $\mathcal{F}$ denote the 2D discrete Fourier transform and $\tilde{\epsilon}_{\alpha,\alpha'} \equiv \mathcal{F}(\boldsymbol{\epsilon})_{\alpha,\alpha'}$ denote a Fourier mode of $\boldsymbol{\epsilon}$. Then eqn. (2.10) becomes a simple scalar equation,

$$\tilde{\epsilon}^l_{\alpha,\alpha'} \approx (\lambda_{\alpha,\alpha'} \chi_{q^*})^{l-l_0} [\tilde{\epsilon}^{l_0}_{\text{d}}]_{\alpha,\alpha'} + (\lambda_{\alpha,\alpha'} \chi_{c^*})^{l-l_0} [\tilde{\epsilon}^{l_0}_{\text{o.d.}}]_{\alpha,\alpha'} \,, \tag{2.11}$$

with $\lambda_{\alpha,\alpha'} = \mathcal{F}(\mathcal{A})^*_{\alpha,\alpha'}$. Thus, the linearized dynamics of convolutional neural networks decouple into independently-evolving Fourier modes that evolve near the fixed point at frequency-dependent rates.

2.1.4. Fixed-point analysis

The stability of the fixed point Σ^* is determined by whether nearby points move closer or farther from Σ^* under the dynamics described by eqn. (2.9). Eqn. (2.11) shows that this condition depends on the whether the quantities $\lambda_{\alpha,\alpha'} \chi_{q^*}$ and $\lambda_{\alpha,\alpha'} \chi_{c^*}$ are less than or greater than one.

Since $\mathcal{A}$ is a diagonal matrix, the eigenvalues $\lambda_{\alpha,\alpha'}$ have a specific structure. In particular, the set of eigenvalues is comprised of n copies of the 1D discrete Fourier transform of the diagonal entries of $\mathcal{A}$. Furthermore, since the diagonal entries of $\mathcal{A}$ are non-negative and sum to one, their Fourier coefficients have absolute value no larger than one and the zero-frequency coefficient is equal to one; see Figure 4 for the full distribution in the case of 2D convolutions. It follows that the fixed point Σ^* will be stable if and only if $\chi_{q^*} < 1$ and $\chi_{c^*} < 1$.

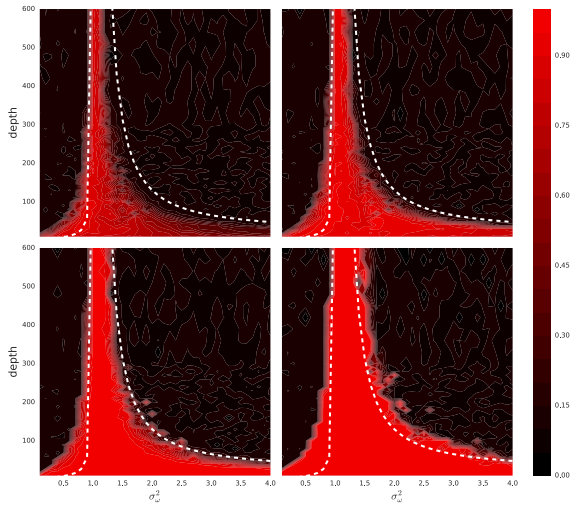

Figure 2. Mean field theory predicts the maximum trainable depth for CNNs. For fixed bias variance $\sigma_b^2 = 2 \times 10^{-5}$, the heat map shows the training accuracy on MNIST obtained for a given depth L network and weight variance σ_w, after (a) 500, (b) 2,500, (c) 10,000, and (d) 100,000 training steps. Also plotted (white dashed line) is a multiple ($6\xi_c$) of the characteristic depth scale governing convergence to the fixed point.

These stability conditions are precisely the ones found to govern fully-connected networks (Poole et al., 2016; Schoenholz et al., 2017). Moreover, the fixed point matrix Σ^* is also the same as in the fully-connected case. Together, these observations imply that the entire fixed-point structure of the convolutional case is identical to that of the fully-connected case. In particular, based on the results of (Poole et al., 2016), we can immediately conclude that the (σ_w, σ_b) hyperparameter plane is separated by the line $\chi_1 = 1$ into an ordered phase with $c^* = 1$ in which all pixels approach the same value, and a chaotic phase with $c^* < 1$ in which the pixels become decorrelated with one another; see the SM for a review of this phase diagram analysis.

2.1.5. Depth scales of signal propagation

We now assume that the conditions for a stable fixed point are met, i.e. $\chi_{q^*} < 1$ and $\chi_{c^*} < 1$, and we consider the rate at which the fixed point is approached. As in (Schoenholz et al., 2017), it is convenient to additionally assume $\chi_{q^*} < \chi_{c^*}$ so that the dynamics in the diagonal subspace can be neglected. In this case, eqn. (2.11) can be rewritten as

$$\tilde{\epsilon}^l_{\alpha,\alpha'} \approx e^{-(l-l_0)/\xi_{\alpha,\alpha'}} [\tilde{\epsilon}_{\text{o.d.}}]^{l_0}_{\alpha,\alpha'} \,, \tag{2.12}$$

where $\xi_{\alpha,\alpha'} = -1/\log(\chi_{c^*} \lambda_{\alpha,\alpha'})$ are depth scales governing the convergence of the different modes. In particular, we expect signals corresponding to a specific Fourier mode

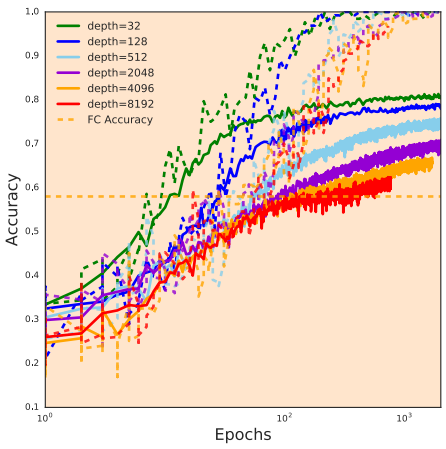

Figure 3. Test (solid) and training (dashed) curves of CNNs with different depths initialized critically using orthogonal kernels on CIFAR-10. Training accuracy reaches 100% for all these curves (except for 8192, which was stopped early) but generalization performance degrades with increasing depth, likely because of attenuation of spatially non-uniform modes. The Delta-Orthogonal initialization in Fig. 1 addresses this reduction in test performance with increasing depth.

$f_{\alpha,\alpha'}$ to be able to travel a depth commensurate to $\xi_{\alpha,\alpha'}$ through the network. Thus, unlike fully-connected networks which exhibit only a single depth scale, convolutional networks feature a hierarchy of depth scales.

Recalling that $\lambda_{\alpha,n-\alpha} = 1$, it follows that $\xi_c \equiv \xi_{\alpha,n-\alpha} = -1/\log \chi_{c^*}$, which is identical to the depth scale governing signal propagation through fully-connected networks. It follows from (Schoenholz et al., 2017) that when $\chi_1 = 1$, $\xi_{\alpha,n-\alpha}$ diverges and thus convolutional networks can propagate signals arbitrarily far through the $f_{\alpha,n-\alpha}$ modes. Since $|\lambda_{\alpha,\alpha'}| < 1$ for $\alpha' \neq n - \alpha$, these are the only modes through which signals can propagate without attenuation. Finally, we note that the $f_{\alpha,n-\alpha}$ modes correspond to perturbations that are spatially uniform along the cyclic diagonals of the covariance matrix. The fact that all signals with additional spatial structure attenuate for large depth suggests that deep critical convolutional networks behave quite similarly to fully-connected networks, which also cannot propagate spatially-structured signals.

2.1.6. Non-uniform kernels

The similarities between signal propagation in convolutional neural networks and fully-connected networks in the limit of large depth are surprising. A consequence may be that the performance of very deep convolutional networks degrades as the signal is forced to propagate along modes with minimal spatial structure. Indeed, Fig. 3 shows that the generalization performance decreases with depth, and that for very large depth it barely surpasses the performance of a fully-connected network.

If increased spatial uniformity is the problem, eqn. (2.12) holds the solution. In order for *all* modes to propagate without attenuation, it is necessary that $\lambda_{\alpha,\alpha'} = 1$ for all α, α'. In fact, it is easy to show that the distribution of $\{\lambda_{\alpha,\alpha'}\}$ can be modified by allowing for spatial non-uniformity in the variance of the weights within the kernel. To this end, we introduce a non-negative vector $v = (v_\beta)_{\beta \in ker}$ chosen such that $\sum_\beta v_\beta = 1$, and initialize the weights of the network according to $w^l_{ij}(\beta) \sim \mathcal{N}(0, \sigma_w^2 v_\beta / c)$. Each choice of v will induce a new dynamical equation analogous to eqn. (2.3) (see SM),

$$\mathbf{\Sigma}^{l+1} = \mathcal{A}_v \star \mathcal{C}(\mathbf{\Sigma}^l)\,, \tag{2.13}$$

where $\mathcal{A}_v = \mathrm{diag}(v)$. It follows directly from the previous analysis that the linearized dynamics of eqn. (2.13) will be identical to the dynamics of eqn (2.3), only now with $\lambda_{\alpha,\alpha'} = \mathcal{F}(\mathcal{A}_v)^*_{\alpha,\alpha'}$. By the same argument presented in Section 2.1.3, the set of eigenvalues is now comprised of n copies of the 1D Fourier transform of v. As a result, it is possible to control the depth scales over which different modes of the signal can propagate through the network by changing the variance vector v. We will return to this point in section 2.4.

2.2. Back-propagation of signal

We now turn our attention to the back-propgation of error signals through a convolutional network. Let E denote the loss and $\delta^l_j(\alpha)$ the back-propagated signal at layer l, channel j and spatial location α, i.e.,

$$\delta^l_j(\alpha) = \frac{\partial E}{\partial h^l_j(\alpha)}. \tag{2.14}$$

The recurrence relation is given by

$$\delta^l_j(\alpha) = \sum_{i \in chn} \sum_{\beta \in ker} \delta^{l+1}_i(\alpha - \beta)\omega^{l+1}_{ji}(\beta)\phi'(h^l_j(\alpha)).$$

As in (Schoenholz et al., 2017), we additionally make the assumption that the weights used during back-propagation are drawn independently from the weights used in forward propagation, in which case the random variables $\{\delta^l_j\}_{j \in chn}$ are independent for each l. The covariance matrices $\tilde{\mathbf{\Sigma}}^l \equiv \mathbb{E}\left[\delta^l_j (\delta^l_j)^T\right]$ back-propagate according to,

$$\tilde{\mathbf{\Sigma}}^l_{\alpha,\alpha'} = \sum_{\beta \in ker} v_\beta \tilde{\mathbf{\Sigma}}^{l+1}_{\alpha-\beta,\alpha'-\beta} \cdot \sigma_w^2 \mathbb{E}_{\boldsymbol{h} \sim \mathcal{N}(0,\mathbf{\Sigma}^l)}[\phi'(h_\alpha)\phi'(h_{\alpha'})]\,. \tag{2.15}$$

We are primarily interested in the diagonal of $\tilde{\mathbf{\Sigma}}^l$, which measures the variance of back-propagated signals. We will also assume $l > l_0$ (see section 2.1.3) so that $\mathbf{\Sigma}^l$ is well-approximated by $\mathbf{\Sigma}^*$. In this case,

$$\tilde{\mathbf{\Sigma}}^l_{\alpha,\alpha} \approx \chi_1 \sum_{\beta \in ker} v_\beta \tilde{\mathbf{\Sigma}}^{l+1}_{\alpha-\beta,\alpha-\beta}\,, \tag{2.16}$$

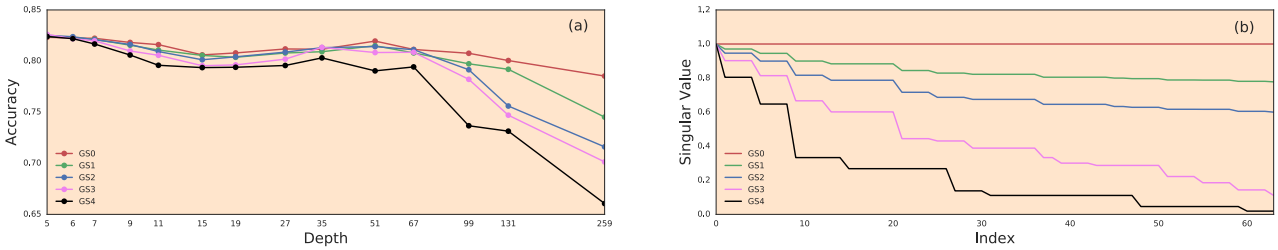

Figure 4. Test performance, as a function of depth, is correlated with the singular value distribution (SVD) of the generalized averaging operator $(\mathcal{A}_v \star)$ (see eqn. (2.13)). (a) Initialized critically, we examine the test accuracy of CNNs with different depths and with Gaussian initialization of different non-uniform variance vectors. We "deform" the variance vector from a delta function (red) to a uniformly distributed one (black). Starting from depth 35, we see the test accuracy curve also "deforms" from the red one to the black one. (b) The SVD of $(\mathcal{A}_v \star)$ for the selected variance vectors. The x-axis represents the index of a singular value, with a total of 64 singular values (each has 64 copies) for each variance vector. See Section 3.3 for details.

where we used eqn. (2.7). Therefore we find that, $\tilde{\mathbf{\Sigma}}^l_{\alpha,\alpha} \sim \chi_1^{L-l} \tilde{\mathbf{\Sigma}}^L_{\alpha,\alpha}$, where L is the total depth of the network. As in the fully-connected case, $\chi_1 = 1$ is a necessary condition for gradient signals to neither explode nor vanish as they back-propagate through a convolutional network. However, as discussed in (Pennington et al., 2017; 2018), this is not always a sufficient condition for trainability. To further understand backward signal propagation, we need to push our analysis beyond mean field theory.

2.2.1. Beyond Mean Field Theory

We have observed that the quantity χ_1 is crucial for determining signal propagation in CNNs, both in the forward and backward directions. As discussed in (Poole et al., 2016), χ_1 equals the the mean squared singular value of the Jacobian $\boldsymbol{J}^l$ of the layer-to-layer transition operator. Beyond just the second moment, higher moments and indeed the whole distribution of singular values of the entire end-to-end Jacobian $\boldsymbol{J} = \prod_l \boldsymbol{J}^l$ are important for ensuring trainability of very deep fully-connected networks (Pennington et al., 2017; 2018). Specifically, networks train well when their input-output Jacobians exhibit *dynamical isometry*, namely the property that the entire distribution of singular values is close to 1.

In fact, we can adopt the entire analysis of (Pennington et al., 2017; 2018) into the convolutional setting with essentially no modification. The reason stems from the fact that, because convolution is a linear operator, it has a matrix representation, $\boldsymbol{W}^l$, which appears in the end-to-end Jacobian in precisely the same manner as do the weight matrices in the fully-connected case. In particular, $\boldsymbol{J} = \prod_{l=1}^{L} \boldsymbol{D}^l \boldsymbol{W}^l$, where $\boldsymbol{D}^l$ is the diagonal matrix whose diagonal elements contain the vectorized representation of derivatives of post-activation neurons in layer l. Roughly speaking, since this is the same expression as in (Pennington et al., 2017; 2018), the conclusions found in that work regarding dynamical isometry apply equally well in the convolutional setting.

The analysis of Pennington et al. (2017; 2018) reveals that the singular values of $\boldsymbol{J}$ depends crucially on the distribution of singular values of $\boldsymbol{W}^l$ and $\boldsymbol{D}^l$. In particular, to achieve dynamical isometry, all of these matrices should be close to orthogonal. As in the fully-connected case, the singular values of $\boldsymbol{D}^l$ can be made arbitrarily close to 1 by choosing a small value for q^* and by using an activation function like $\tanh$ that is smooth and linear near the origin. In the convolutional setting, the matrix representation of the convolution operator $\boldsymbol{W}^l$ is a $c \times c$ block matrix with $n \times n$ circulant blocks. Note that in the large c limit, $n/c \to 0$ and the relative size of the blocks vanishes. Therefore, if the weights are i.i.d. random variables, we can invoke universality results from random matrix theory to conclude its singular value distribution converges to the Marcenko-Pastur distribution; see Fig. S3 in the SM. As such, we find that CNNs with i.i.d. weights cannot achieve dynamical isometry. We address this issue in the next section.

2.3. Orthogonal Initialization for CNNs

In (Pennington et al., 2017; 2018), it was observed that dynamical isometry can lead to dramatic improvements in training speed, and that achieving these favorable conditions requires orthogonal weight initializations. While the procedure to generate random orthogonal weight matrices in the fully-connected setting is well-known, it is less obvious how to do so in the convolutional setting, and at first sight it is not at all clear whether it is even possible. We resolve this question by invoking a result from the wavelet literature (Kautsky & Turcajov, 1994) and provide an explicit construction. We will focus on the *two-dimensional* convolution here and begin with some notation.

Definition 2.1. *We say $K \in \mathbb{R}^{\mathbb{k}\times\mathbb{k}\times c_{in}\times c_{out}}$ is an orthogonal kernel if for all $x \in \mathbb{R}^{n\times n\times c_{in}}$, $\|K * x\|_2 = \|x\|_2$.*

Definition 2.2. *Consider the block matrices $B = \{B_{i,j}\}_{0\le i,j\le p-1} \in \mathbb{R}^{pn\times pn}$ and $C = \{C_{i,j}\}_{0\le i,j\le q-1} \in \mathbb{R}^{qn\times qn}$, with constituent blocks $B_{i,j} \in \mathbb{R}^{n\times n}$ and $C_{i,j} \in$*

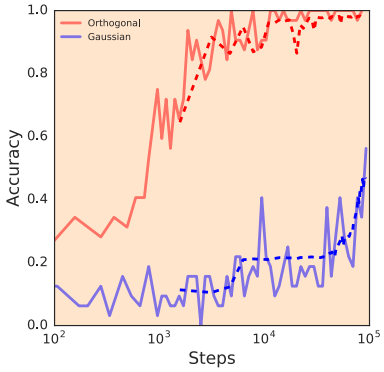

Figure 5. Orthogonal initialization leads to faster training in CNNs. Training (solid lines) and test curves for a 4,000-layer CNN trained using orthogonal (red) and Gaussian (blue) initializations with identical weight variance.

$\mathbb{R}^{n\times n}$. *Define the block-wise convolution operator* $\square$ *by,*

$$[B\square C]_{i,j} = \sum_{i',j'} B_{i',j'} C_{i-i',j-j'}, \qquad (2.17)$$

where the out-of-range matrices are taken to be zero.

Algorithm 1 shows how to construct orthogonal kernels for 2D convolutions of size $\Bbbk \times \Bbbk \times c_{in} \times c_{out}$ with $c_{in} \leq c_{out}$. One can employ the same method to construct kernels of higher (or lower) dimensions. This new initialization method can dramatically boost the learning speed of deep CNNs; see Fig. 5 and Section 3.2.

2.4. Delta-Orthogonal Initialization

In Section 2.1.5 it was observed that, in contrast to fully-connected networks, CNNs have multiple depth scales controlling propagation of signals along different Fourier modes. Even at criticality, for generic variance-averaging vectors v, the majority of these depth scales are finite. However, there does exist one special averaging vector for which all of the depth scales are infinite: a one-hot vector, i.e. $v_i = \delta_{k,i}$. This kernel places all of its variance in the spatial center of the kernel and zero variance elsewhere. In this case, the eigenvalues $\lambda_{\alpha,\alpha'}$ are all equal to 1 and all depth scales diverge, implying that signals can propagate arbitrarily far along all Fourier modes.

If we combine this special averaging vector with the orthogonal initialization of the previous section, we obtain a powerful new initialization scheme that we call Delta-Orthogonal Initialization. Matrices of this type can be generated from Algorithm 1 with $\Bbbk = 1$ and padding with appropriate zeros or directly from Algorithm 2 in the SM.

In the following sections, we demonstrate experimentally that extraordinarily deep convolutional networks can be trained with these initialization techniques.

Algorithm 1 2D orthogonal kernels for CNNs, available in TensorFlow via the ConvolutionOrthogonal initializer.

Input: $\Bbbk$ kernel size, c_{in} number of input channels, c_{out} number of output channels.
Return: a $\Bbbk \times \Bbbk \times c_{in} \times c_{out}$ tensor K.
Step 1. Let K be the $1 \times 1 \times c_{out} \times c_{out}$ tensor such that $K[0,0] = I$, where I is the $c_{out} \times c_{out}$ identity matrix.
Step 2. Repeat the following $(\Bbbk - 1)$ times:
Randomly generate two orthogonal projection matrices P and Q of size $c_{out} \times c_{out}$ and set (see eqn. (2.17))

$$K \leftarrow K\square \begin{bmatrix} PQ & P(1-Q) \\ (1-P)Q & (1-P)(1-Q) \end{bmatrix}.$$

Step 3. Randomly generate a $c_{in} \times c_{out}$ matrix H with orthonormal rows and for $i = 0, \ldots, \Bbbk - 1$ and $j = 0, \ldots, \Bbbk - 1$, set $K[i,j] \leftarrow HK[i,j]$.
Return K.

3. Experiments

To support the theoretical results built up in Section 2, we trained a large number of very deep CNNs on MNIST and CIFAR-10 with $\tanh$ as the activation function. We use the following *vanilla* CNN architecture. First we apply three $3 \times 3 \times c$ convolutions with strides 1, 2 and 2 in order to increase the channel size to c and reduce the spatial dimension to 7×7 (or 8×8 for CIFAR-10), and then a block of d $3 \times 3 \times c$ convolutions with d varying from 2 to $10,000$. Finally, an average pooling layer and a fully-connected layer are applied. Here $c = 256$ when $d \leq 256$ and $c = 128$ otherwise. To maximally support our theories, we applied *no* common techniques (including learning rate decay). Note that the early downsampling is necessary from a computational perspective, but it does diminish the maximum achievable performance; e.g. our best achieved test accuracy with downsampling was 82% on CIFAR-10. We performed an additional experiment training a 50 layers network without downsampling. This resulted in a test accuracy of 89.90%, which is comparable to the best performance on CIFAR-10 using a $\tanh$ architecture that we were able to find (89.82%, (Mishkin & Matas, 2015)).

3.1. Trainability and Critical Initialization

The analysis in Section 2.1 gives a prediction for precisely which initialization hyperparameters a CNN will be trainable. In particular, we predict that the network ought to be trainable provided $L \lesssim \xi_c$. To test this, we train a large number of convolutional neural networks on MNIST with depth varying between $L = 10$ and $L = 600$ and with weights initialized with $\sigma_w^2 \in [0,4]$. In Fig. 2 we plot – using a heatmap – the training accuracy obtained by these networks after different numbers of steps. Additionally we

overlay the depth scale predicted by our theory, ξ_c. We find strikingly good agreement between our theory of random networks and the results of our experiments.

3.2. Orthogonal Initialization and Ultra-deep CNNs

We argued in Section 2.2.1 that the input-output Jacobian of CNNs with i.i.d. weights will become increasingly ill-conditioned as the number of layers grows. On the other hand, orthogonal weight initializations can achieve dynamical isometry and dramatically boost the training speed. To verify this, we train a 4,000-layer CNN on MNIST using a critically-tuned Gaussian weight initialization and the orthogonal initialization scheme developed in Section 2.3. Fig. 5 shows that the network with Gaussian initialization learns slowly (test and training accuracy is below 60% after $90,000$ steps, about 60 epochs). In contrast, orthogonal initialization learns quickly with test accuracy above 60% after only 1 epoch, and achieves 95% after $10,000$ steps or about 7 epochs.

3.3. Multi-dimensional Signal Propagation

The analysis in Section 2.1.3 and Section 2.1.6 suggest that CNNs initialized with kernels with spatially uniform variance may suffer a degradation in generalization performance as the depth increases. Fig. 3 shows the learning curves of CNNs on CIFAR-10 with depth varying from 32 to 8192. Although the orthogonal initialization enables even the deepest model to reach 100% training accuracy, the test accuracy decays as the depth increases with the deepest mode generalizing only marginally better than a fully-connected network.

To test whether this degradation in performance may be the result of attenuation of spatially non-uniform signals, we trained a variety of models on CIFAR-10 whose kernels were initialized with spatially non-uniform variance. According to the analysis in Section 2.1.6, changing the shape of this non-uniformity controls the depth scales over which different Fourier components of the signal can propagate through the network. We examined five different non-uniform critical Gaussian initialization methods. The variance vectors v were chosen in the following way: GS0 refers to the one-hot delta initialization for which the eigenvalues $\lambda_{\alpha,\alpha'}$ are all equal to 1. GS1, GS2 and GS3 are obtained by interpolating between GS0 and GS4, which is the uniform variance initialization.

Each variance vector has exactly 8×8 singular values, plotted in Fig. 4(b) in descending order. Note that from GS0 to GS4, the singular values become more poorly-conditioned (the distribution becomes more concentrated around 0). Fig. 4(a) shows that the relative fall-off of generalization performance with depth follows the same pattern: the more poorly-conditioned the singular values the worse the model generalizes. These observations suggest that salient information may be propagating along multiple Fourier modes.

3.4. Training 10,000-layers: Delta-Orthogonal Initialization.

Our theory predicts that an ultra-deep CNNs can train faster and perform better if critically initialized using Delta-Orthogonal kernels. To test this theory, we train CNNs of 1,250, 2,500, 5,000 and 10,000 layers on both MNIST and CIFAR-10 (Fig. 1). All these networks learn surprisingly quickly and, remarkably, the learning time measured in number of training epochs is independent of depth. Furthermore, our experimental results match well with the predicted benefits of this initialization: 99% test accuracy on MNIST for a 10,000-layer network, and 82% on CIFAR-10. To isolate the benefits of the Delta-Orthogonal init, we also train a 2048-layer CNN (Fig. 3) using the spatially-uniform orthogonal initialization proposed in Section 2.3; the testing accuracy is about 70%. Note that the test accuracy using (spatially uniform) Gaussian (non-orthogonal) initialization is already below 70% when the depth is 259.

4. Discussion

In this work, we developed a theoretical framework based on mean field theory to study the propagation of signals in deep convolutional neural networks. By examining the necessary conditions for signals to flow both forward and backward through the network without attenuation, we derived an initialization scheme that facilitates training of vanilla CNNs of unprecedented depths. We presented an algorithm for the generation of random orthogonal convolutional kernels, an ingredient that is necessary to enable dynamical isometry, i.e. good conditioning of the network's input-output Jacobian. In contrast to the fully-connected case, signal propagation in CNNs is intrinsically multi-dimensional – we showed how to decompose those signals into independent Fourier modes and how to promote uniform signal propagation across them. By leveraging these various theoretical insights, we demonstrated empirically that it is possible to train vanilla CNNs with 10,000 layers or more.

Our results indicate that we have removed all the major fundamental obstacles to training arbitrarily deep vanilla convolutional networks. In doing so, we have layed the groundwork to begin addressing some outstanding questions in the deep learning community, such as whether depth alone can deliver enhanced generalization performance. Our initial results suggest that past a certain depth, on the order of tens or hundreds of layers, the test performance for vanilla convolutional architecture saturates. These observations suggest that architectural features such as residual connections and batch normalization are likely to play an important role in defining a good model class, rather than simply enabling efficient training.

Acknowledgements

We thank Xinyang Geng, Justin Gilmer, Alex Kurakin, Jaehoon Lee, Hoang Trieu Trinh, and Greg Yang for useful discussions and feedback.

References

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537, 2011.

Daniely, A., Frostig, R., and Singer, Y. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, pp. 2253–2261. Curran Associates, Inc., 2016.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Hanin, B. and Rolnick, D. How to start training: The effect of initialization and architecture. *arXiv preprint arXiv:1803.01719*, 2018.

Hayou, S., Doucet, A., and Rousseau, J. On the selection of initialization and activation function for deep neural networks. *arXiv preprint arXiv:1805.08266*, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016a.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pp. 630–645. Springer, 2016b.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

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

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

Karakida, R., Akaho, S., and Amari, S.-i. Universal Statistics of Fisher Information in Deep Neural Networks: Mean Field Approach. *ArXiv e-prints*, June 2018.

Kautsky, J. and Turcajov, R. A matrix approach to discrete wavelets. In Chui, C. K., Montefusco, L., and Puccio, L. (eds.), *Wavelets: Theory, Algorithms, and Applications*, volume 5 of *Wavelet Analysis and Its Applications*, pp. 117 – 135. Academic Press, 1994.

Kim, Y. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Mishkin, D. and Matas, J. All you need is a good init. *CoRR*, abs/1511.06422, 2015. URL http://arxiv.org/abs/1511.06422.

Pennington, J., Schoenholz, S., and Ganguli, S. Resurrecting the sigmoid in deep learning through dynamical isometry: theory and practice. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, pp. 4788–4798. Curran Associates, Inc., 2017.

Pennington, J., Schoenholz, S. S., and Ganguli, S. The emergence of spectral universality in deep networks. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain*, pp. 1924–1932, 2018. URL http://proceedings.mlr.press/v84/pennington18a.html.

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., and Ganguli, S. Exponential expressivity in deep neural networks through transient chaos. *NIPS*, 2016.

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

Schoenholz, S. S., Gilmer, J., Ganguli, S., and Sohl-Dickstein, J. Deep Information Propagation. *ICLR*, 2017.

Schoenholz, S. S., Pennington, J., and Sohl-Dickstein, J. A correspondence between random neural networks and statistical field theory. *arXiv preprint arXiv:1710.06570*, 2017.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe,

D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 01 2016.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

Yang, G. and Schoenholz, S. Mean field residual networks: On the edge of chaos. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, pp. 2865–2873. Curran Associates, Inc., 2017.

Yang, G. and Schoenholz, S. S. Deep mean field theory: Layerwise variance and width variation as methods to control gradient explosion, 2018. URL https://openreview.net/forum?id=rJGY8GbR-.