Frequency Bias in Neural Networks for Input of Non-Uniform Density

Ronen Basri [1] **Meirav Galun** [1] **Amnon Geifman** [1] **David Jacobs** [2] **Yoni Kasten** [1] **Shira Kritchman** [1]

Abstract

Recent works have partly attributed the generalization ability of over-parameterized neural networks to frequency bias – networks trained with gradient descent on data drawn from a uniform distribution find a low frequency fit before high frequency ones. As realistic training sets are not drawn from a uniform distribution, we here use the Neural Tangent Kernel (NTK) model to explore the effect of variable density on training dynamics. Our results, which combine analytic and empirical observations, show that when learning a pure harmonic function of frequency κ, convergence at a point $\mathbf{x} \in \mathbb{S}^{d-1}$ occurs in time $O(\kappa^d/p(\mathbf{x}))$ where $p(\mathbf{x})$ denotes the local density at $\mathbf{x}$. Specifically, for data in $\mathbb{S}^1$ we analytically derive the eigenfunctions of the kernel associated with the NTK for two-layer networks. We further prove convergence results for deep, fully connected networks with respect to the spectral decomposition of the NTK. Our empirical study highlights similarities and differences between deep and shallow networks in this model.

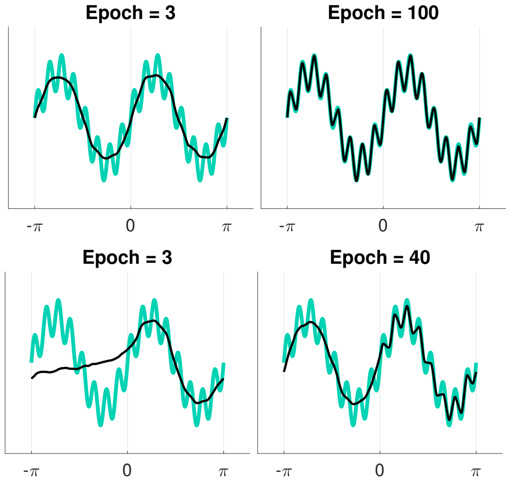

Figure 1. Frequency bias under uniform (top) and non-uniform (bottom) distributions. The light cyan line represents the target function which is composed of the sum of low and high frequency functions. The thin black line represents the network output. Top: when training data is distributed uniformly, low frequency (left) is learned before high frequency (right). Bottom: with non-uniform distribution (positive region is dense, negative is sparse), a good low frequency fit for the low density region is obtained only after 40 epochs, but by then the network fits most of the high frequency component of the target function at the dense region.

1. Introduction

A key question in understanding the success of neural networks is: what makes over-parameterized networks generalize so well, avoiding solutions that overfit the training data? In search of an explanation, a number of recent papers (Farnia et al., 2018; Rahaman et al., 2019; Xu et al., 2019) have suggested that training with gradient descent (GD) (as well as SGD) yields a frequency bias – in early epochs training a neural net yields a low frequency fit to the target function, while high frequencies are learned only in later epochs, if they are needed to fit the data (see Figure 1(top)).

This frequency bias has been carefully analyzed in the case of over-parameterized, two-layer networks with rectified linear unit (ReLU) activation, when only the first layer is trained. The dynamics of GD in this case was shown to match the dynamics of GD for the corresponding Neural Tangent Kernel (NTK) (Arora et al., 2019b; Du et al., 2019; Jacot et al., 2018). Assuming the training data is distributed uniformly on a hypersphere, the NTK matrix forms a convolution on the sphere. Its eigenvectors consist of the spherical harmonic functions (Basri et al., 2019; Xie et al., 2017), and its eigenvalues shrink monotonically with frequency, yielding longer convergence times for high frequency components. Specifically, for training data on the circle, high frequencies are learned quadratically slower than low frequencies, and this frequency-dependent gap increases exponentially with dimension (Basri et al., 2019; Bietti & Mairal, 2019; Cao et al., 2019).

All this previous work assumed that training data is dis-

[1] Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel [2] Department of Computer Science, Univeristy of Maryland, College Park, MD, USA. Correspondence to: Ronen Basri <ronen.basri@weizmann.ac.il>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

tributed uniformly. However, realistic training datasets are distributed with a non-uniform density. A natural question therefore is to what extent frequency bias is exhibited for such datasets? Below we provide evidence that frequency bias interacts with density. We show that in any region of the input space with locally constant density, low frequencies are still learned much faster than high frequencies, but the rate of learning also depends linearly on the density. This phenomenon is demonstrated in Figure 1(bottom).

Our paper contains both theoretical and empirical results. We first focus on analyzing the NTK model for two-layer networks with ReLU activation and 2D input, normalized to lie on the unit circle, allowing for input drawn from a non-uniform density that is piecewise constant. For this model we derive closed form expressions for its eigenfunctions and eigenvalues. These eigenfunctions contain functions of piecewise constant local frequency, with higher frequencies where the density of the training data is higher. This implies that we learn high frequency components of a target function faster in regions of higher density. This also allows us to prove that a pure 1-dimensional sine function of frequency κ is learned in time $O(\kappa^2/p^*)$, where p^* denotes the minimum density in the input space. Our experiments illustrate these results and further suggest that for input on a $d-1$-dimensional hypersphere, spherical harmonics are learned in time $O(\kappa^d/p^*)$.

We next examine the NTK for deep, fully connected (FC) networks. We first prove that given a target function $y(\mathbf{x})$, training networks of finite width with GD converges to y at a speed that depends on the projection of y over the eigenvectors of the NTK, extending previous results proved for two-layer networks (Arora et al., 2019b; Cao et al., 2019). We further show that for uniform data the eigenfunctions of NTK consist of the spherical harmonics. We complement these observations with several empirical findings. (1) We show that for uniformly distributed data the eigenvalues decay with frequency, suggesting that frequency bias exists also in deep FC networks. Moreover, similar to two-layer networks, a pure harmonic function of frequency κ is learned in time $O(\kappa^d)$ asymptotically in κ. However, deeper networks appear to learn frequencies of lower k faster than shallow ones. (2) For training data drawn from non-uniform densities the eigenfunctions of NTK appear indistinguishable from those obtained for two-layer networks, indicating that with deep nets learning a harmonic of frequency κ should also require $O(\kappa^d/p^*)$ iterations.

Our results have several implications. First, we extend results that have been proven for training data with a uniform density to the more realistic case of non-uniform density, also extending results for shallow networks to deep, fully connected networks. These results support the idea that real neural networks have a frequency bias that can explain their ability to avoid overfitting. Second, while it is not surprising that networks fit functions of all frequencies more slowly in regions with low data density, we demonstrate that this is the case and quantify this effect. Our results have an interesting implication for training that uses early stopping to regularize the solution. Suppose the signal one wishes to fit is low frequency, and it is corrupted by high frequency noise. Because a network learns low frequency signals more slowly in regions of low density, by the time the signal is learned in these regions, the network will also have learned high frequency components of the noise in regions of high density. This is illustrated in Figure 1(bottom).

2. Prior work

Many recent papers attempt to explain the generalization ability of overparameterized nets. Perhaps the most convincing relate overparameterized networks to kernel methods. (Jacot et al., 2018) identified a family of kernels, termed Neural Tangent Kernels, and showed that neural networks behave like these kernels, in the limit of infinite widths. Related work investigated variants of these kernels, showing that networks of finite, albeit very large widths converge to zero training error almost always and deriving generalization bounds for such networks. These analyses were applied to two-layer networks (Bach, 2017; Bietti & Mairal, 2019; Du et al., 2019; Vempala & Wilmes, 2018; Xie et al., 2017), multilayer perceptrons (i,e. fully connected), residual and convolutional networks (Allen-Zhu et al., 2018; 2019; Arora et al., 2019a; Huang & Yau, 2019; Lee et al., 2019).

However, these kernel models have been criticised for requiring unrealistically wide networks. Additionally, it is still debated if such linear dynamics (referred to as “lazy training”) fully explain the performance of neural networks. Recent theoretical and empirical results suggest that NTK models still somewhat underperform common nonlinear networks (Arora et al., 2019a; Chizat et al., 2019; Novak et al., 2019; Woodworth et al., 2019).

Other work suggested that networks are biased to learn simple functions, and in particular that GD proceeds by first fitting a low frequency function to the target function, and only fits the higher frequencies in later epochs (Rahaman et al., 2019; Xu et al., 2019; Farnia et al., 2018). Additional work (Bach, 2017; Basri et al., 2019; Bietti & Mairal, 2019; Cao et al., 2019) proved the existence of frequency bias in NTK models of two-layer networks and derived convergence rates of training as a function of target frequency. All of these works assumed that training data is distributed uniformly. (Canu & Elisseef, 1999) proposed loss functions that allow higher frequency fit in regions where training data is dense, and only low frequency fit in the sparse regions. Our results suggest that such a penalization may be implicitly enforced in NTK models.

Classical work on kernel methods acknowledged the importance of understanding the eigenfunctions and eigenvalues of kernels for non-uniform data distributions, but focused mainly on bounding the difference between the empirical kernel matrix and the theoretical kernel for the given distribution (e.g., (Shawe-Taylor et al., 2005; Williams & Seeger, 2000)). (Liang & Lee, 2013) derived analytic expressions for the eigenfunctions of polynomial kernels. (Goel & Klivans, 2017) investigated the Gram matrix of the data distribution and showed that sufficiently fast decay of its eigenvalues allows learnability by neural networks. We are unaware of works that derive analytic expressions for the eigenfunctions of NTK under non-uniform distributions.

3. Preliminaries

We consider in this work NTK models for fully connected neural networks with rectified linear unit (ReLU) activations. These kernels are defined through the following formula

$$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbb{E}_{\mathbf{w}\sim\mathcal{I}} \left\langle \frac{\partial f(\mathbf{x}_i, \mathbf{w})}{\partial \mathbf{w}}, \frac{\partial f(\mathbf{x}_j, \mathbf{w})}{\partial \mathbf{w}} \right\rangle, \quad (1)$$

where $f(\mathbf{x}, \mathbf{w})$ is the output of an infinite width network for point $\mathbf{x} \in \mathbb{R}^d$ with parameters $\mathbf{w}$, $\mathbf{x}_i$ and $\mathbf{x}_j$ are any two training points, and the expectation is over the possible initializations of $\mathbf{w}$, denoted $\mathcal{I}$ (usually normal distribution).

We first consider a two layer network with bias:

$$f(\mathbf{x}; \mathbf{w}) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \sigma(\mathbf{w}_r^T \mathbf{x} + b_r), \quad (2)$$

where $\|\mathbf{x}\| = 1$ (denoted $\mathbf{x} \in \mathbb{S}^{d-1}$) is the input, the vector $\mathbf{w}$ includes the weights and bias terms of the first layer, denoted respectively $W = [\mathbf{w}_1, ..., \mathbf{w}_m] \in \mathbb{R}^{d\times m}$ and $\mathbf{b} = [b_1, ..., b_m]^T \in \mathbb{R}^m$, as well as the weights of the second layer, denoted $\mathbf{a} = [a_1, ..., a_m]^T \in \mathbb{R}^m$. σ denotes the ReLU function, $\sigma(x) = \max(x, 0)$. Bias is important in the case of two-layer networks since without bias such networks are non-universal and cannot express harmonic functions of odd frequencies except frequency 1 (Basri et al., 2019).

We then consider deep fully-connected networks with $L + 1 > 2$ layers. For such networks we forgo the bias since our empirical results (Section 5) indicate that they are universal even without bias. These networks are expressed as

$$f(\mathbf{x}; \mathbf{w}) = W^{(L+1)} \cdot \sqrt{\frac{c_\sigma}{d_L}} \sigma \left(W^{(L)} \cdot \sqrt{\frac{c_\sigma}{d_{L-1}}} \sigma \left(W^{(L-1)} \cdots \sqrt{\frac{c_\sigma}{d_1}} \sigma \left(W^{(1)} \mathbf{x} \right) \right) \right), \quad (3)$$

where $\mathbf{x} \in \mathbb{R}^{d_1}$, $\|\mathbf{x}\| = 1$, the parameters $\mathbf{w}$ include $W^{(L+1)}, W^{(L)}, ..., W^{(1)}$, where $W^{(l)} \in \mathbb{R}^{d_l \times d_{l-1}}$, $W^{(L+1)} \in \mathbb{R}^{1\times d_L}$, and $c_\sigma = 1/\left(\mathbb{E}_{z\sim\mathcal{N}(0,1)}[\sigma(z)^2]\right) = 2$.

We assume that n training points are sampled i.i.d. from an arbitrary distribution $p(\mathbf{x})$ on the hypersphere and that each sample $\mathbf{x}_i$ is supplied with a target value $y_i \in \mathbb{R}$ from an unknown function $y_i = g(\mathbf{x}_i)$. Our theoretical derivations further assume that $p(\mathbf{x})$ is piecewise constant. The network is trained to minimize the ℓ_2 loss

$$\Phi(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2. \quad (4)$$

using gradient descent (GD).

For our analysis, to simplify the NTK expressions, in the case of a two-layer network we only train the weights and bias of the first layer (as in (Arora et al., 2019b; Du et al., 2019)). We initialize these weights from a normal distribution $\mathbf{w}_r^{(0)}, b_r^{(0)} \sim \mathcal{N}(0, \tau^2 I)$. We further initialize a_r from a uniform distribution on $\{-1, 1\}$ and keep those weights fixed. In the case of deep networks we train all the weights, initializing by $\mathbf{w} \sim \mathcal{N}(0, I)$.

We next provide expressions for the corresponding neural tangent kernels. For a two-layer network with bias where only the first layer weights are trained the corresponding NTK takes the form (Basri et al., 2019)

$$k(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{4\pi} (\mathbf{x}_i^T \mathbf{x}_j + 1)(\pi - \arccos(\mathbf{x}_i^T \mathbf{x}_j)). \quad (5)$$

When the training data is distributed uniformly, this kernel forms a convolution operator, and so its eigenfunctions are the spherical harmonics on the hypersphere $\mathbb{S}^{d-1}$ (or Fourier series when $d = 2$). The eigenvalues shrink at the rate of $O(1/\kappa^d)$, where κ denotes the frequency of the spherical harmonic functions. Gradient descent training of a target function composed of a pure harmonic requires a number of iterations that is inversely proportional to the corresponding eigenvalue, i.e., $O(\kappa^d)$. (Bach, 2017; Basri et al., 2019; Bietti & Mairal, 2019; Cao et al., 2019; Xie et al., 2017)

For a deep FC network the NTK with L layers, denoted $\Theta_\infty^{(L)}(\mathbf{x}_i, \mathbf{x}_j)$, is expressed by the following recursion (Arora et al., 2019a; Jacot et al., 2018)

$$\Theta_\infty^{(h)}(\mathbf{x}_i, \mathbf{x}_j) = \Theta_\infty^{(h-1)}(\mathbf{x}_i, \mathbf{x}_j)\dot{\Sigma}^{(h)}(\mathbf{x}_i, \mathbf{x}_j) + \Sigma^{(h)}(\mathbf{x}_i, \mathbf{x}_j), \quad (6)$$

where for $h \in [L]$

$$\Theta_\infty^{(0)}(\mathbf{x}_i, \mathbf{x}_j) = \Sigma^{(0)}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$$

$$\Lambda^{(h)}(\mathbf{x}_i, \mathbf{x}_j) = \begin{bmatrix} \Sigma^{(h-1)}(\mathbf{x}_i, \mathbf{x}_i) & \Sigma^{(h-1)}(\mathbf{x}_i, \mathbf{x}_j) \\ \Sigma^{(h-1)}(\mathbf{x}_j, \mathbf{x}_i) & \Sigma^{(h-1)}(\mathbf{x}_j, \mathbf{x}_j) \end{bmatrix}$$

$$\Sigma^{(h)}(\mathbf{x}_i, \mathbf{x}_j) = c_\sigma \mathbb{E}_{(u,v)\sim\mathcal{N}(0,\Lambda^{(h)})}[\sigma(u)\sigma(v)]$$

$$\dot{\Sigma}^{(h)}(\mathbf{x}_i, \mathbf{x}_j) = c_\sigma \mathbb{E}_{(u,v)\sim\mathcal{N}(0,\Lambda^{(h)})}[\dot{\sigma}(u)\dot{\sigma}(v)].$$

Here $\dot{\sigma}(\cdot)$ denotes the step function (i.e., the derivative of the ReLU function). The covariance matrices have the form

$\Lambda = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ with $|\rho| \leq 1$, and the expectations have the following closed form expressions

$$\mathbb{E}_{(u,v)\sim\mathcal{N}(0,\Lambda^{(h)})}[\sigma(u)\sigma(v)] = \frac{\rho(\pi - \arccos(\rho)) + \sqrt{1-\rho^2}}{2\pi}$$

$$\mathbb{E}_{(u,v)\sim\mathcal{N}(0,\Lambda^{(h)})}[\dot{\sigma}(u)\dot{\sigma}(v)] = \frac{\pi - \arccos(\rho)}{2\pi}.$$

4. The eigenfunctions of NTK for two-layer networks for non-uniform distributions

We begin by investigating the NTK model for two-layer networks when the training is drawn from a non-uniform distribution. Focusing first on 1D target functions $y(\mathbf{x}) : \mathbb{S}^1 \to \mathbb{R}$ and a piecewise constant data distribution $p(\mathbf{x})$, we derive explicit expressions for the eigenfunctions and eigenvalues of NTK. This allows us to prove that learning a one-dimensional function of frequency κ requires $O(\kappa^2/p^*)$ iterations, where p^* denotes the minimal density in $p(x)$. We complement these theoretical derivations with experiments with functions in higher dimensions, which indicate that learning functions of frequency κ in $\mathbb{S}^{d-1}$ requires $O(\kappa^d/p^*)$ iterations.

Consider the NTK model described in (5), which corresponds to an infinitely wide, two-layer network for which only the first layer is trained. Suppose that n training data points are sampled from a non-uniform, piecewise constant distribution $p(\mathbf{x})$ on the circle, $\mathbf{x} \in \mathbb{S}^1$. We then form an $n \times n$ matrix H^p whose entries for samples $\mathbf{x}_i$ and $\mathbf{x}_j$ consist of $H^p_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, with k as defined in (5). Following (Arora et al., 2019b), the convergence rates of GD for such a network will depend on the eigen-system of H^p. To analyze this eigen-system, we consider the limit of H^p as the number of points goes to infinity. In this limit the eigen-system of H^p approaches the eigen-system of the kernel $k(\mathbf{x}_i, \mathbf{x}_j)p(\mathbf{x}_j)$, where the eigenfunctions $f(x)$ satisfy the following equation (Shawe-Taylor et al., 2005; Williams & Seeger, 2000),

$$\int_{\mathbb{S}^1} k(\mathbf{x}_i, \mathbf{x}_j)p(\mathbf{x}_j)f(\mathbf{x}_j)d\mathbf{x}_j = \lambda f(\mathbf{x}_i). \tag{7}$$

This is a homogeneous Fredholm Equation of the second kind with the non-symmetric polar kernel $k(\mathbf{x}_i, \mathbf{x}_j)p(\mathbf{x}_j)$. The existence of the eigenfunctions with real eigenvalues is established by symmetrizing the kernel. Let $\tilde{k}(\mathbf{x}_i, \mathbf{x}_j) = p^{1/2}(\mathbf{x}_i)k(\mathbf{x}_i, \mathbf{x}_j)p^{1/2}(\mathbf{x}_j)$ and $g(\mathbf{x}) = p^{1/2}(\mathbf{x})f(\mathbf{x})$. Multiplying (7) by $p^{1/2}(\mathbf{x}_i)$ yields

$$\int_{\mathbb{S}^d} \tilde{k}(\mathbf{x}_i, \mathbf{x}_j)g(\mathbf{x}_j)d\mathbf{x}_j = \lambda g(\mathbf{x}_i), \tag{8}$$

implying the eigenfunctions exist and λ is real.

We next parameterize the unit circle by angles, and denote by x, z any two angles. We can therefore express (7) as

$$\int_{x-\pi}^{x+\pi} k(x,z)p(z)f(z)dz = \lambda f(x), \tag{9}$$

where the kernel in (5) expressed in terms of angles reads

$$k(x,z) = \frac{1}{4\pi}(\cos(x-z)+1)(\pi - |x-z|). \tag{10}$$

Both $p(x)$ and $f(x)$ are periodic with a period of 2π since x lies on the unit circle.

4.1. Explicit expressions for the eigenfunctions

Below we solve (9) and derive an explicit expression for the eigenfunctions $f(x)$. Our derivation assumes that $p(x)$ is piecewise constant. While this assumption limits the scope of our solution, empirical results suggest that when $p(x)$ changes continuously the eigenfunctions are modulated continuously, consistently with our solution. We summarize:

Proposition 1. *Let $p(x)$ be a piecewise constant density function on $\mathbb{S}^1$. Then the eigenfunctions in (9) take the general form*

$$f(x) = a(p(x))\cos\left(\frac{q}{Z}\Psi(x) + b(p(x))\right), \tag{11}$$

where q is integer, $\Psi(x) = \int_{-\pi}^{x}\sqrt{p(\tilde{x})}d\tilde{x}$ and $Z = \frac{1}{2\pi}\Psi(\pi)$.

Note that if $p(x) = p_j$ is constant in a connected region $R_j \subseteq \mathbb{S}^1$, then (11) can be written as

$$f(x) = a_j \cos\left(\frac{q\sqrt{p_j}x}{Z} + b_j\right), \forall x \in R_j. \tag{12}$$

In other words, over the region R_j, this is a cosine function with frequency proportional to $\sqrt{p_j}$. A plot of eigenfunctions for a piecewise constant distribution is shown in Fig. 2.

The proof of the proposition relies on a lemma, proved in supplementary material, stating that the solution to (9) satisfies the following second order ordinary differential equation (ODE)

$$f''(x) = -\frac{p(x)}{\pi\lambda}f(x). \tag{13}$$

In a nutshell, the lemma is proved by applying a sequence of six derivatives to (9) with respect to x, along with some algebraic manipulations, yielding a sixth order ODE for $f(x)$. Assuming that $p(x)$ is piecewise constant simplifies the ODE. Then (13) is obtained by restricting $p(x)$ to have a period of π, but this restriction can be lifted by preprocessing the data in a straightforward way without changing the function that needs to be learned.

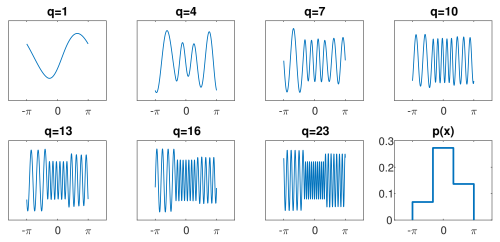

Figure 2. For the NTK of a two-layer network with bias we plot its eigenfunctions (in a decreasing order of eigenvalues) under a non-uniform data distribution in $\mathbb{S}^1$. Here we used a density composed of three constant regions with $p(x) \in 3/(2\pi)\{1/7, 2/7, 4/7\}$ (bottom right plot).

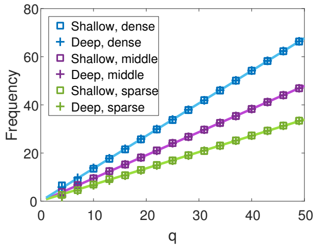

Figure 3. The local frequency in the eigenfunctions within each of the three constant region densities in Figure 2, plotted for both a two-layer and deep (depth=10) networks (marked respectively by squares and plus signs). Measurements are obtained by applying FFT to each region. The measurements are in close match to our formula (12) (solid line).

Eq. (13) has the following general solutions

$$f(x) = Ae^{i\frac{\Psi(x)}{\sqrt{\pi\lambda}}x} + Be^{-i\frac{\Psi(x)}{\sqrt{\pi\lambda}}x}, \tag{14}$$

such that the derivative of Ψ is $\Psi'(x) = \sqrt{p(x)}$, resulting in real eigenfunctions of the form

$$f(x) = a(p(x))\cos\left(\frac{\Psi(x)}{\sqrt{\pi\lambda}}x + b(p(x))\right). \tag{15}$$

As with the uniform distribution, due to periodic boundary conditions there is a countable number of eigenvalues, and those can be determined (up to scale) using the known eigenvalues for the uniform case (Basri et al., 2019). With this we obtain

$$\lambda = \begin{cases} Z^2\left(\frac{1}{2\pi^2} + \frac{1}{8}\right) & q = 0 \\ Z^2\left(\frac{1}{\pi^2} + \frac{1}{8}\right) & q = 1 \\ \frac{Z^2(q^2+1)}{\pi^2(q^2-1)^2} & q \geq 2 \ \text{ even} \\ \frac{Z^2}{\pi^2 q^2} & q \geq 2 \ \text{ odd.} \end{cases} \tag{16}$$

q is integer, and there is one eigenfunction for $q = 0$ and two eigenfunctions for every $q > 0$. Figure 4 shows a plot of the eigenvalues computed for various densities.

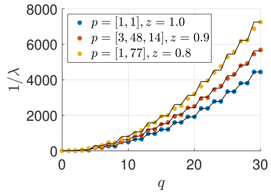

Figure 4. The kernel eigenvalues for several distributions, including uniform (in blue), three bins with $p = 3/(130\pi)[3, 48, 14]$ (red), and two bins with $p = 2/(78\pi)[1, 77]$. The formula (marked by the solid lines) closely matches the eigenvalues H^p computed numerically using $50K$ points.

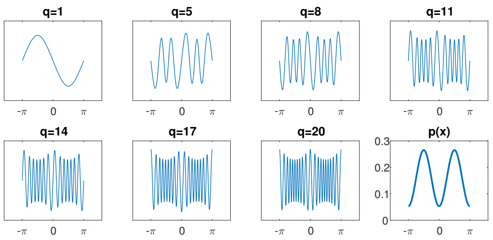

Figure 5. For the NTK of a two-layer network we plot the empirical eigenvectors of H^p for a continuous distribution, $p(x) = \frac{3\cos(2x+\pi)+4.5}{9\pi}$ (bottom right).

The amplitudes and phase shifts are determined by requiring the eigenfunctions to be continuous and differentiable everywhere. We show in supplementary material that for two neighboring regions, $j, j+1$ it holds that if $p_j \leq p_{j+1}$ then the ratio of the amplitudes is bounded (tightly) for different values of p_j and p_{j+1} as follows:

$$1 \leq \frac{a_j}{a_{j+1}} \leq \sqrt{\frac{p_{j+1}}{p_j}}. \tag{17}$$

Figure 2 shows the eigenvectors and eigenvalues for an example of a piecewise constant distribution. It can be seen that each eigenfunction consists of a piecewise sine function; i.e., the eigenfunctions in every region where $p(x)$ is constant form pure sine functions with frequency that changes from one region to the next. As we inspect eigenfunctions with decreasing eigenvalues we find, as our theory shows (see Figure 3), that the frequencies increase in all regions, but for all eigenfunctions they maintain constant ratios that are equal to the ratios between the square roots of the corresponding densities. Finally, Figure 5 shows the eigenvectors of H^p for a continuous distribution, showing similar behaviour to our analytic expressions.

4.2. Time to convergence

Determining the eigenfunctions and eigenvalues of the NTK allows us to predict the number of iterations needed to learn target functions and to understand effects due to varying densities. To understand this we consider target functions of the form $g(x) = \cos(\kappa x)$ where x is drawn from a piecewise constant distribution $p(x)$ on $\mathbb{S}^1$. Denote by $R_j \subseteq \mathbb{S}^1$, $1 \le j \le l$ the regions of constant density. Loosely speaking (see Figure 6), for each region R_j we expect $g(x)$ will correlate well with one eigenfunction (and perhaps to additional ones, but with less energy). Of these, the region corresponding to the lowest density should correlate with an eigenfunction with the smallest eigenvalue. This eigenvalue, which depends on both the target frequency κ and the density $p(x)$ within that region, will determine the number of iterations to convergence. This is summarized in the following theorem.

Theorem 1. *Let $p(x)$ be a piecewise constant distribution on $\mathbb{S}^1$. Denote by $u^{(t)}(x)$ the prediction of the network at iteration t of GD. For any $\delta > 0$ the number of iterations t needed to achieve $\|g(x) - u^{(t)}(x)\| < \delta$ is $\tilde{O}(\kappa^2/p^*)$, where p^* denotes the minimal density of $p(x)$ in $\mathbb{S}^1$ and $\tilde{O}(.)$ hides logarithmic terms.*

Proving this theorem is complicated by the fact that (1) the frequency of the target function may not be exactly represented in the eigenfunctions of the kernel, due to the discrete number of eigenfunctions, and (2) the eigenfunctions restricted to any given region R_j are not orthogonal. These two properties may result in non-negligible correlations of $g(x)$ with eigenfunctions of yet smaller eigenvalues. Therefore, to prove Theorem 1 we first inspect the projections of $g(x)$ onto the eigenfunctions corresponding to such small eigenvalues and prove a bound on this tail. Subsequently we use this bound to prove the convergence rate in the theorem. The proofs are provided in the supplementary material.

In Figure 7 we used the target function $g(x) = \sin(\kappa x)$ for different values of κ to train a 2-layer network. The data was sampled from a non-uniform distribution with three constant regions of densities $3/(2\pi)(1/7, 2/7, 4/7)$. It can be seen that runtime increases for each region in proportion to κ^2, and the network converged faster at denser regions (in proportion to $p(x)$).

4.3. Higher dimension

Deriving analytic expressions for data drawn from a non-uniform distribution in higher dimension, i.e., in $\mathbb{S}^{d-1}$, $d > 2$ is challenging and is left for future work. However, simulation experiments lead us to conjecture that the main properties in $\mathbb{S}^1$ hold also in higher dimension, i.e., (1) the eigenfunctions for piecewise constant distributions resemble concatenated patches of spherical harmonics, (2) the frequencies of these harmonics change with density, and increase monotonically as the respective eigenvalues become smaller, and (3) learning a harmonic function of frequency k should require $O(k^d/p^*)$ iterations.

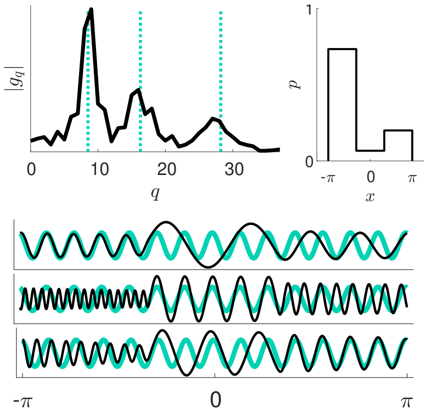

Figure 6. Illustration of Thm. 1. For a piecewise constant density with three regions (top right), a function $g(x) = \sin(14x)$ (in green, bottom plots) is projected onto the eigenfunctions of k (three of which are shown with black curves in the bottom plots), producing coefficients g_q (top left). This produces three peaks around the points predicted by our theory (marked by the dotted vertical lines), which correspond to high correlation of $g(x)$ with one of the three regions for the appropriate three basis functions (bottom row).

Figure 8 shows an example plot of eigenfunctions in $\mathbb{S}^2$ with a density function that is constant in each hemisphere. We further used harmonic functions of different frequencies to train a two-layer network with bias. Figure 9 shows convergence time as a function of frequency. As conjectured, for each region convergence time increases roughly in proportion to k^3, and convergence in different regions is linearly faster with density.

5. Deep networks

We next extend our discussion to NTK models of deep, fully connected networks. We first prove that the eigenvec-

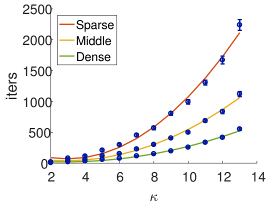

Figure 7. Convergence times as a function of the target frequency κ for a two-layer network trained with data drawn from a non-uniform distribution in $\mathbb{S}^1$. We used the distribution of Figure 2, which is composed of three regions of constant density with a ratio of 1:2:4. For each region R_j the network converges at time proportional to κ^2/p_j, as is indicated by the three quadratic curves fit to the data points. In addition, the median ratios between our measurements for the three regions are 1:1.96:3.89, in close fit to the distribution.

tors of NTK indeed characterize the convergence of GD of highly overparmeterized networks of *finite* width. We then empirically investigate the eigenvectors and eigenvalues of NTK for data drawn from either uniform or non-uniform distributions and show convergence times for pure sine and harmonic target functions.

We begin by showing that the eigenvectors of NTK characterize the dynamics of overparameterized FC networks of finite width. Our theorem extends Thm. 4.1 in (Arora et al., 2019b) (see also (Cao et al., 2019)), which has dealt with two-layer networks, to deep nets. Consider a FC network of depth L and width m in each layer, and suppose the network is trained with n pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$. Denote the vector of target values by $\mathbf{y} = (y_1, ..., y_n)$ and the network predictions for these values at time t by $\mathbf{u}^{(t)}$. In our theorem, Thm. 2, we use a slightly different model than the model stated above (3). First, we assume that the first and last layers are initialized and then held fixed throughout training, and the last layer is initialized randomly $\sim \mathcal{N}(0, \tau^2 I)$. The NTK for this training data is summarized in an $n \times n$ matrix H^∞, whose entries are set to $H_{ij}^\infty = k(\mathbf{x}_i, \mathbf{x}_j)$ where k is defined in (1). Let $\mathbf{v}_i$ and λ_i respectively denote the eigenvectors of H^∞ and their corresponding eigenvalues. The next theorem establishes that the convergence rate of training this deep (finite width) network depends on the decomposition of the target values $\mathbf{y}$ over the eigenvectors of H^∞.

Theorem 2. *For any $\epsilon \in (0, 1]$ and $\delta \in (0, O(\frac{1}{L})]$, let $\tau = \Theta(\frac{\epsilon\hat{\delta}}{n})$, $m \geq \Omega\left(\frac{n^{24}L^{12}\log^5 m}{\delta^8\tau^6}\right)$, $\eta = \Theta\left(\frac{\delta}{n^4L^2m\tau^2}\right)$. Then, with probability of at least $1 - \hat{\delta}$ over the random initialization after t GD iterations we have that*

$$\|\mathbf{y} - \mathbf{u}^{(t)}\| = \sqrt{\sum_{i=1}^{n}(1-\eta\lambda_i)^{2t}(\mathbf{v}_i^T\mathbf{y})^2} \pm \epsilon. \tag{18}$$

The proof is provided in the supplementary material. Below, we give a brief proof sketch. First, we show that for any number of layers and at any iteration t the following relation holds

$$\mathbf{u}^{(t+1)} - \mathbf{y} = (I - \eta H(t))(\mathbf{u}^{(t)} - \mathbf{y}) + \epsilon(t), \tag{19}$$

where $H_{ij}(t) = \left\langle \frac{\partial f(\mathbf{x}_i, \mathbf{w}(t))}{\partial \mathbf{w}}, \frac{\partial f(\mathbf{x}_j, \mathbf{w}(t))}{\partial \mathbf{w}} \right\rangle$, and the residual $\epsilon(t)$ due to the GD steps is relatively small. Then, based on several results due to (Allen-Zhu et al., 2019; Arora et al., 2019a), we show that $H(t)$ can be approximated by H^∞, yielding, by applying recursion to (19)

$$\mathbf{u}^{(t)} - \mathbf{y} = (I - \eta H^\infty)^t(\mathbf{u}^{(0)} - \mathbf{y}) + \xi(t). \tag{20}$$

where $\|\xi(t)\| \leq O(\epsilon)$. Next we show that under the setting of τ, $\|\mathbf{u}^{(0)}\| \leq O(\epsilon)$. Finally, by applying the spectral decomposition to H^∞ we obtain (18).

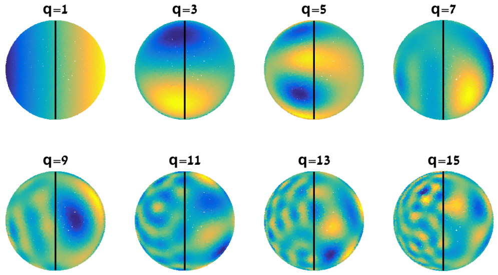

Figure 8. The eigenfunctions of NTK for a two-layer network with bias for data drawn from a non-uniform distribution from $\mathbb{S}^2$. The left and right hemispheres each have constant density with a ratio of 12:1.

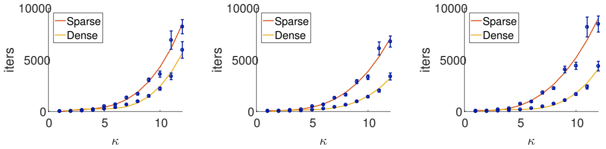

Figure 9. Convergence times as a function of the target harmonic frequency κ for a two-layer network trained with data drawn from a non-uniform distribution in $\mathbb{S}^2$. In each plot the sphere was divided into 2 halves, with density ratios (from left to right) of 1:2, 1:3, 1:4. The plot shows a cubic fit to the measurements. The median ratios between our measurements for the three subplots are 1.76, 2.45 and 2.99, undershooting our conjectured ratios. We believe this is due to sensitivity of experiments on $\mathbb{S}^2$ to sampling.

Our next aim is to compute the eigenvectors and eigenvalues of NTK matrices for deep networks. This, together with Theorem 2, will allow us to derive convergence rates for different target functions. Toward that aim we observe that the NTK kernel $k(\mathbf{x}_i, \mathbf{x}_j)$ is a function of the inner product of its arguments. This can be concluded from its recursive definition in (6), since $\Sigma^{(0)}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$; both $\Sigma^{(h)}(\mathbf{x}_i, \mathbf{x}_j)$ and $\dot{\Sigma}^{(h)}(\mathbf{x}_i, \mathbf{x}_j)$ are (scaled) expectations over random variables drawn from a zero normal distribution and whose covariance, by recursion, is a function of the inner product $\mathbf{x}_i^T\mathbf{x}_j$. Consequently, the kernel decomposes over the zonal spherical harmonics in $\mathbb{S}^{d-1}$ (or Fourier series in $\mathbb{S}^1$), and for training data drawn from the uniform distribution the corresponding kernel matrix forms a convolution.

Figure 10 shows for the NTK of depth 10 that indeed the eigenvectors in $\mathbb{S}^1$ are the Fourier series. We note that despite the lack of bias terms all frequencies are included. The eigenvalues decrease monotonically with frequency, indicating that the network should learn low frequency functions faster than high frequency ones. Moreover, as Figures 11 and 12 show, regardless of depth, when trained with a function of frequency κ overparameterized networks converge respectively at the asymptotic speed of $O(\kappa^2)$ and $O(\kappa^3)$ for uniform data in $\mathbb{S}^1$ and $\mathbb{S}^2$. Interestingly, however, the

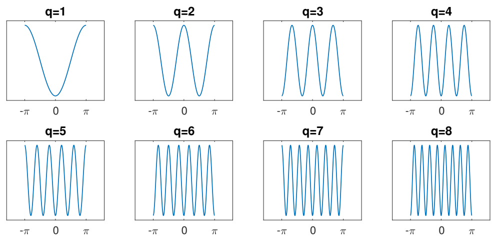

Figure 10. The eigenfunctions of NTK for a deep network (depth 10) for the uniform distribution in $\mathbb{S}^1$. The eigenvectors are arranged according to a descending order of their corresponding eigenvalues.

eigenvalues of NTK reveal a difference in the way deep and shallow networks treat low frequencies in the target function, as is reflected by the plots in Figure 12. Each line of one color represents the log of the eigenvalues for one network and the lines are ordered from shallow to deep in ascending order. The local slope of these lines indicate the speed of convergence for the corresponding frequencies. Asymptotically all the lines become parallel as the frequency κ increases, implying that the asymptotic convergence times should be equal for all depths. However, for the low frequencies the lines corresponding to deeper networks are flatter than those corresponding to shallow networks. This flatter slope indicates that the frequency bias for such frequencies is smaller, implying that deep networks learn frequencies, e.g., 6-10, almost as fast as 1-5, while this is not true for shallow networks.

Finally, for data drawn from a non-uniform distribution the eigenfunctions of NTK for deep networks appear to be indistinguishable from those obtained for two-layer networks. Figure 3 shows a plot of the local frequencies obtained with NTK for a network of depth 10. It can be seen that the local frequencies are identical to those obtained with NTK for a two-layer network. The eigenvalues are similar to those obtained with the uniform density, up to a normalizing scale which depends on the distribution. Similarly to the two-layer case, learning a harmonic function of frequency κ is therefore expected to require $O(\kappa^d/p^*)$ iterations.

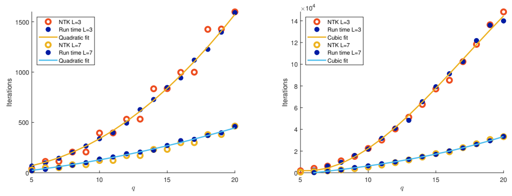

Figure 11. For deep networks (3 and 7 layers) and data drawn from the uniform distribution in $\mathbb{S}^1$ (left) and $\mathbb{S}^2$ (right) we plot training times as a function of target frequency (marked by the solid blue circles). This is compared to the times predicted by the eigenvalues of the corresponding NTK model (red circles).

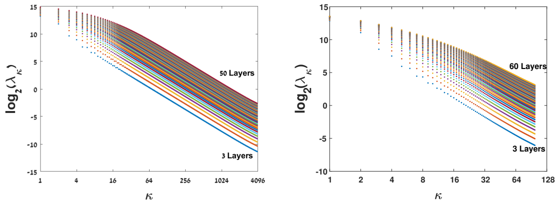

Figure 12. This figure shows a plot of the eigenvalues of NTK for FC networks of different depths with points drawn from a uniform density in $\mathbb{S}^1$ (left) and $\mathbb{S}^2$ (right). The plot is given in log-log scale. Networks of different depths are colored differently. Plots for deeper networks appear higher due to scaling. It can be seen that all curves decrease monotonically, indicating that the eigenvalues decay with frequency. In addition they all become parallel as the frequency κ grows, converging to a slope of -2 for $\mathbb{S}^1$ and -3 for $\mathbb{S}^2$ (fitting the curves in the left plot starting at $\kappa = 50$ yields a slope of 1.94; fitting the right plot starting at $\kappa = 10$ yields a slope of 2.80). This indicates that asymptotically the rate of learning a frequency κ is $O(\kappa^2)$ and $O(\kappa^3)$ respectively regardless of depth. The shallower slope of deep networks on the left part of each plot indicates that middle frequencies are learned faster with deep networks than with shallow ones.

6. Conclusion

The main contribution of our work is to show that insights about neural networks that have been derived with the assumption of uniformly distributed training data also apply, in interesting ways, to more realistic, non-uniform data. Prior work has shown that the Neural Tangent Kernel provides a model of real, overparameterized neural networks that is tractable to analyze and that matches real experiments. Our work shows that NTK has a frequency bias for non-uniform data distributions as well as for uniform ones. This strengthens the case that this frequency bias may play an important role in real neural networks.

We also quantify this frequency bias. We derive an expression for the eigenfunctions of NTK, showing that for piecewise constant data distributions the eigenfunctions consist of piecewise harmonic functions. The frequency of these piecewise functions increases linearly with the square root of the local density of the data. As a consequence, for 1D inputs, networks modeled by NTK learn harmonic functions with a speed that increases quadratically in their frequency and decreases linearly with the local density. Experiments indicate that these results generalize naturally to higher dimensions. These results support the idea that overparameterized networks avoid overfitting because they fit target functions with smooth functions, and are slow to add high frequency components that could overfit.

Acknowledgements

This material is based partly upon work supported by the National Science Foundation under Grant No. DMS1439786 while the authors were in residence at the Institute for Computational and Experimental Research in Mathematics in Providence, RI, during the Computer Vision program. The authors thank the U.S.- Israel Binational Science Foundation, grant number 2018680 and the Quantifying Ensemble Diversity for Robust Machine Learning (QED for RML) program from DARPA for their support of this project.

References

Allen-Zhu, Z., Li, Y., and Song, Z. On the convergence rate of training recurrent neural networks. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 2018.

Allen-Zhu, Z., Li, Y., and Song, Z. A convergence theory for deep learning via over-parameterization. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 242–252, 2019.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R., and Wang, R. On exact computation with an infinitely wide neural net. In *NeurIPS*, 2019a.

Arora, S., Du, S. S., Hu, W., Li, Z., and Wang, R. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. *arXiv preprint arXiv:1901.08584*, 2019b.

Bach, F. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18:1–53, 2017.

Basri, R., Jacobs, D., Kasten, Y., and Kritchman, S. The convergence rate of neural networks for learned functions of different frequencies. In *NeurIPS*, 2019.

Bietti, A. and Mairal, J. On the inductive bias of neural tangent kernels. In *NeurIPS*, 2019.

Canu, M. F. and Elisseef, A. Regularization , kernels and sigmoid netst. In *INSA, Rouen*, 1999.

Cao, Y., Fang, Z., Wu, Y., Zhou, D.-X., and Gu, Q. Towards understanding the spectral bias of deep learning, 2019.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, 2019.

Du, S. S., Zhai, X., Poczos, B., and Singh, A. Gradient descent provably optimizes over-parameterized neural networks. *International Conference on Learning Representations (ICLR)*, 2019.

Farnia, F., Zhang, J., and Tse, D. A spectral approach to generalization and optimization in neural networks. 2018.

Goel, S. and Klivans, A. R. Eigenvalue decay implies polynomial-time learnability for neural networks. In *NIPS*, 2017.

Huang, J. and Yau, H.-T. Dynamics of deep neural networks and neural tangent hierarchy. *arXiv preprint arXiv:1909.08156*, 2019.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 8580–8589, 2018.

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. In *Advances in Neural Information Processing Systems*, pp. 8570–8581, 2019.

Liang, Z. and Lee, Y. Eigen-analysis of nonlinear pca with polynomial kernels. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 6(6):529–544, 2013.

Novak, R., Xiao, L., Bahri, Y., Lee, J., Yang, G., Hron, J., Abolafia, D. A., Pennington, J., and Sohl-Dickstein, J. Bayesian deep convolutional networks with many channels are gaussian processes. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019.

Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., Bengio, Y., and Courville, A. On the spectral bias of neural networks. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5301–5310. PMLR, 2019.

Shawe-Taylor, J., Williams, C. K., Cristianini, N., and Kandola, J. On the eigenspectrum of the gram matrix and the generalization error of kernel-pca. *IEEE Trans. Inf. Theor.*, 51(7):2510–2522, 2005.

Vempala, S. S. and Wilmes, J. Gradient descent for one-hidden-layer neural networks: Polynomial convergence and sq lower bounds. In *COLT*, 2018.

Williams, C. and Seeger, M. The effect of the input density distribution on kernel-based classifiers. In *Proceedings of the 17th International Conference on Machine Learning*, pp. 1159–1166, 2000.

Woodworth, B. E., Gunasekar, S., Lee, J. D., Soudry, D., and Srebro, N. Kernel and deep regimes in overparametrized models. *CoRR*, abs/1906.05827, 2019.

Xie, B., Liang, Y., and Song, L. Diverse neural network learns true target functions. In *International Conference on Artificial Intelligence and Statistics (AISTATS), Fort Lauderdale, Florida*, pp. 1216–1224, 2017.

Xu, Z. J., Zhang, Y., Luo, T., Xiao, Y., and Ma, Z. Frequency principle: Fourier analysis sheds light on deep neural networks. *CoRR*, abs/1901.06523, 2019.