A Convergence Rate for Manifold Neural Networks

Joyce A. Chew
Department of Mathematics
University of California, Los Angeles
Los Angeles, California 90095
Email: joycechew@math.ucla.edu

Deanna Needell
Department of Mathematics
University of California, Los Angeles
Los Angeles, California 90095
Email: deanna@math.ucla.edu

Michael Perlmutter
Department of Mathematics
Boise State University
Boise, Idaho 83725
Email: mperlmutter@boisestate.edu

***Abstract*—High-dimensional data arises in numerous applications, and the rapidly developing field of geometric deep learning seeks to develop neural network architectures to analyze such data in non-Euclidean domains, such as graphs and manifolds. Recent work has proposed a method for constructing manifold neural networks using the spectral decomposition of the Laplace-Beltrami operator. Moreover, in this work, the authors provide a numerical scheme for implementing such neural networks when the manifold is unknown and one only has access to finitely many sample points. They show that this scheme, which relies upon building a data-driven graph, converges to the continuum limit as the number of sample points tends to infinity. Here, we build upon this result by establishing a rate of convergence that depends on the intrinsic dimension of the manifold but is independent of the ambient dimension. We also discuss how the rate of convergence depends on the depth of the network and the number of filters used in each layer.**

***Index Terms*—Geometric Deep Learning, Manifold Learning, Continuum Limits**

I. Introduction

The emerging field of Geometric Deep Learning [1], [2] aims to extend the success of convolutional neural networks to more general domains such as graphs and manifolds. Though only in its infancy, this field has already achieved substantial success. For instance, Graph Neural Networks are used to power both Amazon's product recommender system [3] and Google Maps [4]. However, while there has been a tremendous body of research on the construction of neural networks for graphs [5], there has been relatively little work on the development of *neural networks for manifolds,* except in the case of two-dimensional surfaces [6]. This is in spite of the fact that manifold learning techniques [7], [8] are often used in unsupervised algorithms for representing high-dimensional by capturing the intrinsic structure of the data. Future developments in the development of manifold neural networks could have tremendous applications in the processing of high-dimensional data arising, for example, in single-cell imaging [9]–[11].

As an important step in the development of manifold neural networks, several recent works [12], [13] have introduced CNN-like architectures that use filters based on the eigendecomposition of the Laplace-Beltrami operator and analyzed their stability properties. However, [12] and [13] assume that one knows the manifold which is not realistic for many real-world applications. Instead, one is given access to finitely many sample points $x_1, \ldots, x_n \in \mathbb{R}^D$ and makes the modeling assumption that these sample points lie upon (or near) an unknown d-dimensional Riemannian manifold for some $d \ll D$. In this setup, it is non-trivial to actually implement a neural network since one does not have global knowledge of the manifold. One approach, based on *manifold learning* [8], [14], [15] is to construct a data-driven graph $\mathbf{G}_n$, whose vertices are the sample points $x_1, \ldots, x_n$, and use the eigenvectors and eigenvalues of graph Laplacian to approximate the eigenfunctions and eigenvalues of the Laplace-Beltrami operator.

Recently, in [16], the authors showed that if the graph $\mathbf{G}_n$ is constructed properly, their proposed neural network will converge to a continuum limit as the number of sample points n tends to infinity. At the same time, a different line of work [17] proposed a numerical method for implementing the manifold scattering transform [18], [19], a wavelet-based model of deep learning on manifolds, in the setting where one only has to sample finitely many points and proved convergence to a continuum limit. Moreover, using results from [20] the authors are able to establish a quantitative rate of convergence which depends on the intrinsic dimension d, but not on the ambient dimension D.

The purpose of this paper is to apply results from [17] and [20] to the manifold neural network considered in [16] and establish a quantitative rate of convergence. Notably, as in [17], our rate of convergence will depend on the intrinsic dimension d, but not on the ambient dimension D. We do not know if our rate of convergence is optimal, but it is likely that some dependence on d cannot be avoided. Additionally, we note that our rate also depends on the number of layers L and the number of filters F_l used in each layer. As we shall discuss below, these dependencies will give insights into how to best design manifold neural networks.

A. Notation

We assume that $\mathcal{M}$ is a compact d-dimensional Riemannian manifold without boundary embedded in $\mathbb{R}^D$ for some $D \gg d$ such that vol$(\mathcal{M}) = 1$, and that the points $\{x_i\}_{i=1}^n$ are uniformly sampled from $\mathcal{M}$. We will let $\mathbf{G}_n$ denote a graph with vertices x_i, $1 \leq i \leq n$. We will let $\mathcal{L}$ denote the Laplace-Beltrami operator on $\mathcal{M}$ and let $\mathbf{L}_n$ denote the graph Laplacian

Accepted to Sampling Theory and Applications Conference 2023, to appear in IEEE Xplore.

on $\mathbf{G}_n$. We will let ϕ_i and λ_i denote the eigenfunctions/values of the Laplace Beltrami operator $\mathcal{L}$ and let ϕ_i^n and λ_i^n denote the eigenvectors/values of the graph Laplacian $\mathbf{L}_n$. We will let $\mathbf{P}_n$ be the projection operator, defined for continuous functions $f \in \mathbf{L}^2(\mathcal{M})$ by $[\mathbf{x}_n]_i = [\mathbf{P}_n f]_i = f(x_i)$. We will let $\|\cdot\|_{\mathbf{L}^2(\mathcal{M})}$ and $\|\cdot\|_{\mathbf{L}^2(\mathbf{G}_n)}$ denote the $\mathbf{L}^2$ norms on $\mathcal{M}$ and $\mathbf{G}_n$. However, when no ambiguity exists we may simple write $\|\cdot\|$ to refer to the $\mathbf{L}^2$ norm on the appropriate space.

II. Manifold Neural Networks

Let $\mathcal{M}$ be a compact d-dimensional Riemannian manifold without boundary embedded in $\mathbb{R}^D$ for some $D \gg d$, and let $\mathcal{L}$ be the Laplace-Beltrami operator on $\mathcal{M}$. It is well-known that $\mathcal{L}$ has an orthonormal basis of eigenfunctions $\{\phi_i\}_{i=1}^\infty$ with $\mathcal{L}\phi_i = \lambda_i\phi_i$, $\lambda_i \geq 0$. This implies that for $f \in \mathbf{L}^2(\mathcal{M})$ we may write

$$f = \sum_{i=1}^\infty \widehat{f}(i)\phi_i,$$

where, for $1 \leq i < \infty$, $\widehat{f}(i)$ is the *generalized Fourier coefficient* defined by $\langle f, \phi_i\rangle_{\mathbf{L}^2(\mathcal{M})}$. Since we interpret the $\widehat{f}(i)$ as generalized Fourier coefficients, we will refer to a function f as bandlimited if there are finitely many nonzero $\widehat{f}(i)$ as formalized in the following definition.

Definition 1 (Bandlimited functions)**.** *For $\kappa \geq 0$, a function $f \in \mathbf{L}^2(\mathcal{M})$ is said to be κ-bandlimited if we have $\widehat{f}(i) = 0$ for all $i > \kappa$.*

Motivated by the convolution theorem in real analysis, we will define manifold convolution as multiplication in the Fourier domain. Given a sufficiently nice function $\widehat{h} : [0,\infty) \to \mathbb{R}$, we let $h \in \mathbf{L}^2(\mathcal{M})$ be the function

$$h = \sum_{i=1}^\infty \widehat{h}(\lambda_i)\phi_i$$

and define a spectral convolution operator, $h(\mathcal{L})$ by

$$h(\mathcal{L})f = \sum_{i=1}^\infty \widehat{h}(\lambda_i)\widehat{f}(i)\phi_i. \tag{1}$$

Importantly, we note that since these spectral convolution operators are defined in terms of a function $\widehat{h} : [0,\infty) \to \mathbb{R}$ one may verify that the $h(\mathcal{L})$ does not depend on the choice of the orthonormal basis $\{\phi_i\}_{i=1}^\infty$. (See for example Remark 1 of [17].)

In [16], the authors use this notion of convolution to define an L-layer manifold neural network with F_ℓ filters used in layer ℓ. In this network, one is given an initial signal $f = f_0 \in \mathbf{L}^2(\mathcal{M})$ and then produces a hidden representation of this signal through a multi-layer sequence of linear and non-linear maps. In particular, for $1 \leq \ell \leq L$ and $1 \leq p \leq F_\ell$, they define

$$f_\ell^p = \sigma\left(\sum_{q=1}^{F_{\ell-1}} h_\ell^{pq}(\mathcal{L}) f_{\ell-1}^q\right), \tag{2}$$

where $h_\ell^{pq}(\mathcal{L})$ is a spectral convolution operator defined as in (1) and σ is a pointwise Lipschitz nonlinearity which is non-expansive in the sense that $|\sigma(a)-\sigma(b)| \leq |a-b|$. Admissible choices of σ include ReLu, absolute value, and many others. As in [16], we will write the network compactly as

$$\mathbf{H} = \cup_{\ell=1}^L \{h_\ell^{pq}, 1 \leq q \leq F_{\ell-1}, 1 \leq p \leq F_\ell\}$$

denote the set of all filters and let $\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)$ denote the output of their network. Throughout this paper, we will assume that all of the h_ℓ^{pq} are non-amplifying and Lipschitz as defined below.

Definition 2 (Non-amplifying filters)**.** *A manifold filter $h(\mathcal{L})$ is called non-amplifying if $\|\widehat{h}\|_\infty \leq 1$.*

Definition 3 (Lipschitz filters)**.** *For $C > 0$, a manifold filter $h(\mathcal{L})$ is called C-Lipschitz if for all $a, b \in [0,\infty)$ we have*

$$|\widehat{h}(a) - \widehat{h}(b)| \leq C|a-b|.$$

We note that the condition that $h(\mathcal{L})$ is non-amplifying allows one to verify (using Parseval's Theorem) that $\|h(\mathcal{L})f\|_{\mathbf{L}^2(\mathcal{M})} \leq \|f\|_{\mathbf{L}^2(\mathcal{M})}$ for all $f \in \mathbf{L}^2(\mathcal{M})$ and the condition that h is Lipschitz implies that $h(\mathcal{L})$ is robust to minor perturbations of the eigenvalues.

As in, e.g., [16] and in Section 6 of [17], we will assume that we have a collection of n points $x_1, \ldots, x_n$, which lie on $\mathcal{M}$ and are drawn i.i.d. uniformly at random with respect to the Riemannian volume form, and build a data-driven graph, $\mathbf{G}_n$, whose adjacency matrix $\mathbf{A}_n$ is defined by

$$[\mathbf{A}_n]_{ij} = w_{ij} = \frac{1}{n}\frac{1}{t_n(4\pi t_n)^{d/2}}\exp\left(-\frac{\|x_i - x_j\|_2^2}{4t_n}\right), \tag{3}$$

where t_n is a suitably chosen scale parameter. Given $\mathbf{A}_n$, the diagonal degree matrix $\mathbf{D}_n$ and graph Laplacian $\mathbf{L}_n$ are defined by

$$\mathbf{L}_n = \mathbf{D}_n - \mathbf{A}_n.$$

It is known that the graph Laplacian has an orthonormal basis of eigenvectors ϕ_i^n such that $\mathbf{L}_n\phi_i^n = \lambda_i^n\phi_i^n$, $\lambda_i^n \geq 0$, which allows us to write

$$\mathbf{x} = \sum_{i=1}^n \widehat{\mathbf{x}}_i\phi_i^n, \quad \widehat{\mathbf{x}}_i = \langle \mathbf{x}, \phi_i^n\rangle.$$

In order to implement a discrete network analogous to (2), we will need to define a projection operator $\mathbf{P}_n$ which maps a function f onto its values at the sample points x_i, $1 \leq i \leq n$.

Definition 4 (Projection operators)**.** *For a continuous function $f \in \mathbf{L}^2(\mathcal{M})$, we define its projection onto $\mathbf{G}_n$ to be the vector $\mathbf{P}_n f$ whose entries are given by $[\mathbf{P}_n f]_i = f(x_i)$.*

Given this definition, one can then implement a discrete approximation of (2) by setting $\mathbf{x}_{n,0} = \mathbf{P}_n f$ and defining

$$\mathbf{x}_{n,\ell}^p = \sigma\left(\sum_{q=1}^{F_{\ell-1}} h_\ell^{pq}(\mathbf{L}_n)\mathbf{x}_{n,\ell-1}^q\right), \tag{4}$$

where analogous to (1), $h(\mathbf{L}_n)$ is defined by

$$h(\mathbf{L}_n)\mathbf{x} = \sum_{i=1}^{n} \widehat{h}(\lambda_i)\widehat{\mathbf{x}}_i \phi_i^n.$$

With these definitions, we may now state the following result which is Proposition 1 of [16]. It shows that this discrete approximation converges to the continuum limit under certain assumptions.

Theorem 1 (Proposition 1 of [16])**.** *Assume that the x_i are drawn i.i.d. uniform at random from $\mathcal{M}$ and that f is κ-bandlimited for some finite κ. Let $\mathbf{H} = \cup_{\ell=1}^{L}\{h_\ell^{pq}, 1 \leq q \leq F_{\ell-1}, 1 \leq p \leq F_\ell\}$, and assume that each $h \in \mathbf{H}$ is non-amplifying and C-Lipschitze for some finite $C > 0$. Assume that $\mathbf{G}_n$ is constructed as in (3), and let $\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)$ and $\mathbf{\Phi}(\mathbf{H}, \mathbf{L_n}, \mathbf{P}_n f)$ denote the output of the continuous and discretized manifold neural networks defined in (2) and (4). Then, letting $\mathbf{P_n}\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)$ denote the projection of the output of the continuum network onto $\mathbf{G}_n$, we have*

$$\lim_{n\to\infty} \|\mathbf{\Phi}(\mathbf{H}, \mathbf{L_n}, \mathbf{P}_n f) - \mathbf{P_n}\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)\|_{\mathbf{L}^2(G_n)} = 0$$

where the above limit is in probability.

While Theorem 1 guarantees convergence, it does not tell provide a quantitative rate. However, recent work by Chew et al. [17], provides methods for implementing the manifold scattering transform[1] from finitely many sample points with a quantitative rate. Similar to the method discussed above, the method used in [17] also relies upon building a data-driven graph $\mathbf{G}_n$ although it builds this graph in a slightly different way, defining the adjacency matrix, diagonal degree matrix, and graph Laplacian by

$$\begin{aligned}
[\mathbf{A}_n]_{i,j} &= t_n{}^{-d/2} \exp\left(-\frac{\|x - x'\|_2^2}{t_n}\right), \\
[\mathbf{D}_n]_{i,i} &= \sum_{j=1}^{n} [\mathbf{A}_n]_{i,j}, \\
\mathbf{L}_n &= \frac{1}{nt_n}\left(\mathbf{D}_n - \mathbf{A}_n\right). \qquad (5)
\end{aligned}$$

With the graph defined this way, they are then able to utilize the following theorem, which is a special case of Theorem 5.4 of [20].

Theorem 2 (Theorem 5.4 of [20])**.** *Assume that the points $\{x_i\}_{i=1}^n$ are drawn i.i.d uniformly at random. Let $\kappa > 0$ be fixed. Assume that the first κ eigenvalues of the (true) Laplace-Beltrami operator, $\mathcal{L}$, $\lambda_1, \ldots, \lambda_\kappa$, all have single multiplicity. Construct $\mathbf{G}_n$ as in (5), and let ϕ_i^n and λ_i^n be the eigenvectors and eigenvalues of the data-driven $\mathbf{L}_n$. Assume that $t_n \to 0$ and $n \to \infty$ at a rate where $t_n \sim n^{-2/(d+6)}$. Then, with probability at least $1 - \mathcal{O}\left(\frac{1}{n^9}\right)$, there exist scalars α_k with*

$$|\alpha_k| = 1 + o(1) \qquad (6)$$

such that for all $0 \leq i \leq \kappa$

$$|\lambda_i - \lambda_i^n| = \mathcal{O}\left(n^{-\frac{2}{d+6}}\right)$$

and

$$\|\mathbf{P}_n\phi_i - \alpha_i\phi_i^n\|_2 = \mathcal{O}\left(n^{-\frac{2}{d+6}}\sqrt{\log n}\right),$$

where the constants implied by the big-$\mathcal{O}$ notation depend on κ and the geometry of $\mathcal{M}$.

Remark 1. *As noted earlier, spectral filters of the form (1) do not depend on the choice of eigenbasis. Therefore, when applying Theorem 2 to the manifold neural networks considered here we may assume that the $\alpha_i > 0$ (since otherwise we change the basis by replacing ϕ_i with $-\phi_i$.)*

Therefore, by imitating the proof of Theorem 10 of [17], with probability at least $1 - \mathcal{O}(\frac{1}{n^9})$, we may see

$$\|\mathbf{P}_n\phi_i - \phi_i^n\|_2 = \mathcal{O}\left(n^{-\frac{2}{d+6}}\sqrt{\log n}\right).$$

The authors of [17] also make frequent use of the following Lemma which is a consequence of Hoeffding's inequality.

Lemma 1 (Lemma 5 of [17])**.** *Assume that the points $\{x_i\}_{i=1}^n$ are drawn i.i.d uniformly at random, and let $f, g \in \mathcal{C}(\mathcal{M})$. Then, with probability at least $1 - \frac{2}{n^9}$ we have*

$$|\langle f, g\rangle_{\mathbf{G}_n} - \langle f, g\rangle_{\mathbf{L}^2(\mathcal{M})}| \leq \sqrt{\frac{18\log n}{n}}\|fg\|_\infty.$$

Given Theorem 2 and Lemma 1, the authors of [17] then show that the manifold scattering transform converges at a quantifiable rate informally summarized below in the following theorem. (See Theorems 13 and 14 [17] for a precise statement).

Theorem 3 (Informal)**.** *Under the assumptions of Theorem 2, the manifold scattering transform converges at rate $\mathcal{O}\left(n^{-\frac{2}{d+6}}\sqrt{\log n}\right)$.*

The main result of this paper, discussed in the following section, establishes a similar convergence result for manifold neural networks.

III. Convergence Rates for Manifold Neural Networks

The following theorem is our main result. It builds upon Theorem 1 by providing a quantitative rate of convergence. Its proof is based on combining Theorem 2 with the techniques used to prove Theorem 1. Full details are provided in the supplementary material.

Theorem 4. *Assume that the x_i are drawn i.i.d. uniformly at random from $\mathcal{M}$ and that f is κ bandlimited. Assume that each of the filters $h \in \mathbf{H}$ is non-amplifying and C-Lipschitz, and let $\tilde{C} = \max\{C, 1\}$. Assume that $\mathbf{G}_n$ is constructed as in (5), let $\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)$ and $\mathbf{\Phi}(\mathbf{H}, \mathbf{L_n}, \mathbf{P}_n f)$ denote the output of the continuous and discretized manifold neural networks defined in (2) and (4) and assume that $t_n \to 0$ and $n \to \infty$ at*

[1] The manifold scattering transform is a nonlinear convolutional architecture similar to that discussed here. It differs by assuming that the filters are wavelets with a certain specified form and by not featuring a summation over channels.

a rate where $t_n \sim n^{-2/(d+6)}$. Then with probability at least $1 - \mathcal{O}(\frac{1}{n^9})$, we have that

$$\begin{aligned}&\|\mathbf{\Phi}(\mathbf{H}, \mathbf{L_n}, \mathbf{P}_n f) - \mathbf{P_n}\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)\|_{\mathbf{L}^2(G_n)}\\ &\leq \tilde{C}\mathcal{O}\left(\frac{\sqrt{\log n}}{n^{2/(d+6)}}\right) \max_{\ell,q} \|f_\ell^q\| \sum_{k=1}^{L} \prod_{j=L-k}^{L} F_j\\ &+ \tilde{C}\mathcal{O}\left(\frac{(\log n)^{1/4}}{n^{1/4+2/(d+6)}}\right) \max_{\ell,q} \|f_\ell^q\|_\infty \sum_{k=1}^{L} \prod_{j=L-k}^{L} F_j\end{aligned}$$

when $d \geq 2$. In the case when $d = 1$, we instead have

$$\begin{aligned}&\|\mathbf{\Phi}(\mathbf{H}, \mathbf{L_n}, \mathbf{P}_n f) - \mathbf{P_n}\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)\|_{\mathbf{L}^2(G_n)}\\ &\leq \tilde{C}\mathcal{O}\left(\frac{\sqrt{\log n}}{n^{2/7}}\right) \max_{\ell,q} \|f_\ell^q\| \sum_{k=1}^{L} \prod_{j=L-k}^{L} F_j\\ &+ \tilde{C}\mathcal{O}\left(\frac{\sqrt{\log n}}{n^{1/2}}\right) \max_{\ell,q} \|f_\ell^q\|_\infty \sum_{k=1}^{L} \prod_{j=L-k}^{L} F_j.\end{aligned}$$

In order to interpret Theorem 4, we may make the following observations on the rate of convergence.

- Our bounds depend on the intrinsic dimension d but not on the ambient dimension D, and the rate of convergence with respect to n is the same as that of analogous results in [17]. We also note that our rate is slightly faster than the rate of $\mathcal{O}(n^{-1/(2d+8)})$ which was established for a single filter (satisfying certain assumptions) in Theorem 3 of [21].
- The rate of convergence depends on the L^2 and L^∞ norms of the filters (although for large n the L^2 term dominates). Therefore, when training manifold neural networks, one might want to employ regularization when training the filters.
- The bound depends linearly on the Lipschitz constant of the filters. Again, this suggests that one might want to employ regularization.
- Networks with more filters (unsurprisingly) require more training data. More specifically, if one uses an L layer network with $F_\ell = F$ filters in each layer, then our bound depends on F via

$$\sum_{k=1}^{L} \prod_{j=L-k}^{L} F_j \approx F^{L+1}.$$

Therefore, very deep networks may require large amounts of training data and so it might be better to use a fairly small number of layers unless one has truly massive amounts of data. Interestingly, this in some sense parallels a well-known phenomenon in graph neural networks where performance typically either levels or deteriorates if more than two or three layers are used (although this is likely for a different reason). This is in contrast to Euclidean CNNs where optimal performance is usually obtained with much deeper networks.

IV. Numerical Experiments

We investigate the convergence rate of the proposed MNN with experiments on the two-dimensional sphere, which enables exact evaluation of the continuous MNN along with the discretized MNN. We uniformly sample up to 2^{14} points on the sphere and construct a signal f by

$$f = \sum_{i=1}^{9} \alpha_i \phi_i$$

where $\alpha_i \in \mathbb{R}$ and ϕ_i are the spherical harmonics, which are the eigenfunctions of the Laplace-Beltrami operator on the sphere. Hence, $\widehat{f}(i) = \alpha_i$ and thus we can exactly compute $h(\mathcal{L})f$. We use a single layer with one filter defined by $\widehat{h}(\lambda) = e^{-\lambda}$ and use the absolute value as the nonlinearity σ. In Figure 1, we plot $\|\mathbf{\Phi}(\mathbf{H}, \mathbf{L_n}, \mathbf{P}_n f) - \mathbf{P_n}\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)\|_{\mathbf{L}^2(G_n)}$ against the number of sampled points N, averaged over 100 trials. We observe a convergence rate of about $\mathcal{O}(n^{-0.76})$ which is faster than the rate guaranteed by our Theorem in the case where $d = 2$, $\mathcal{O}(n^{-1/4})$. This suggests that it may be possible to sharpen our rate of convergence in future work, at least in cases such as the sphere where we have additional information about the structure of the eigenfunctions.

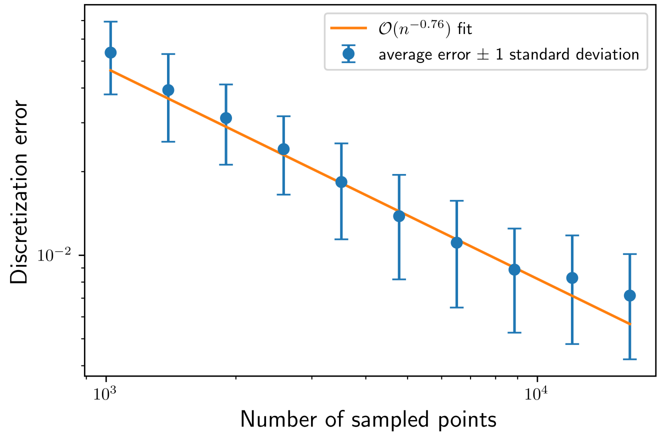

Fig. 1: Errors and log-log fit over 10 logarithmically spaced values of n from 2^{10} to 2^{14} points. Both axes are logarithmically scaled.

V. Conclusions and Future Work

We have established a quantitive rate of convergence for the manifold neural network introduced in [12], [13]. Notably, our rate of convergence depends on the depth of the network and the number of filters used in each layer suggesting that truly deep networks may require massive amounts of training data. This raises an important question for future inquiry. Can one design a manifold neural network whose sampling requirements scale at most linearly with respect to the depth of the network and the number of filters used per layer?

Additionally, we note that both Theorem 1 (restated from [16]) and Theorem 4 assume that the sample points x_i lie exactly on the manifold $\mathcal{M}$ and are sampled i.i.d. uniformly with respect to the Riemannian volume form. We view removing these assumptions as an important line of future inquiry. One possible solution for addressing the first issue is to use techniques from noise-robust manifold learning (see [22] and the references therein) in which one assumes the data has the form $y_i = x_i + \eta_i$ where $x_i \in \mathcal{M}$ and η_i is noise. Similarly, for the second challenge, one could consider methods used in works such as [8], [23], [24] where the data is assumed to be sampled from a non-uniform density. However, adapting these results to our setting is non-trivial because most of these works establish pointwise convergence guarantees whereas our proof techniques require convergence in $\mathbf{L}^2$.

Acknowledgments

Joyce Chew was supported by NSF DGE 2034835. Deanna Needell was supported by NSF DMS 2011140 and NSF DMS 2108479.

References

[1] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond Euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[2] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric deep learning: Grids, groups, graphs, geodesics, and gauges," *arXiv preprint arXiv:2104.13478*, 2021.

[3] Z. Wang and V. N. Ioannidis, "How AWS uses graph neural networks to meet customer needs," 2022.

[4] A. Derrow-Pinion, J. She, D. Wong, O. Lange, T. Hester, L. Perez, M. Nunkesser, S. Lee, X. Guo, B. Wiltshire *et al.*, "Eta prediction with graph neural networks in google maps," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 3767–3776.

[5] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.

[6] J. Masci, D. Boscaini, M. Bronstein, and P. Vandergheynst, "Geodesic convolutional neural networks on riemannian manifolds," in *Proceedings of the IEEE international conference on computer vision workshops*, 2015, pp. 37–45.

[7] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[8] R. R. Coifman and S. Lafon, "Diffusion maps," *Applied and Computational Harmonic Analysis*, vol. 21, pp. 5–30, 2006.

[9] K. R. Moon, D. van Dijk, Z. Wang, W. Chen, M. J. Hirn, R. R. Coifman, N. B. Ivanova, G. Wolf, and S. Krishnaswamy, "Phate: a dimensionality reduction method for visualizing trajectory structures in high-dimensional biological data," *BioRxiv*, vol. 120378, 2017.

[10] D. Van Dijk, R. Sharma, J. Nainys, K. Yim, P. Kathail, A. J. Carr, C. Burdziak, K. R. Moon, C. L. Chaffer, D. Pattabiraman *et al.*, "Recovering gene interactions from single-cell data using data diffusion," *Cell*, vol. 174, no. 3, pp. 716–729, 2018.

[11] E.-a. D. Amir, K. L. Davis, M. D. Tadmor, E. F. Simonds, J. H. Levine, S. C. Bendall, D. K. Shenfeld, S. Krishnaswamy, G. P. Nolan, and D. Pe'er, "visne enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia," *Nature biotechnology*, vol. 31, no. 6, pp. 545–552, 2013.

[12] Z. Wang, L. Ruiz, and A. Ribeiro, "Stability of manifold neural networks to deformations," *arXiv preprint arXiv:2106.03725*, 2021.

[13] ——, "Stability of neural networks on manifolds to relative perturbations," *arXiv preprint arXiv:2110.04702*, 2021.

[14] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[15] ——, "Convergence of Laplacian eigenmaps," in *Advances in Neural Information Processing Systems*, 2007, pp. 129–136.

[16] Z. Wang, L. Ruiz, and A. Ribeiro, "Convolutional neural networks on manifolds: From graphs and back," *arXiv preprint arXiv:2210.00376*, 2022.

[17] J. Chew, M. Hirn, S. Krishnaswamy, D. Needell, M. Perlmutter, H. Steach, S. Viswanath, and H.-T. Wu, "Geometric scattering on measure spaces," *arXiv preprint arXiv:2208.08561*, 2022.

[18] M. Perlmutter, F. Gao, G. Wolf, and M. Hirn, "Geometric scattering networks on compact Riemannian manifolds," in *Mathematical and Scientific Machine Learning Conference*, 2020.

[19] J. Chew, H. Steach, S. Viswanath, H.-T. Wu, M. Hirn, D. Needell, M. D. Vesely, S. Krishnaswamy, and M. Perlmutter, "The manifold scattering transform for high-dimensional point cloud data," in *Topological, Algebraic and Geometric Learning Workshops 2022*. PMLR, 2022, pp. 67–78.

[20] X. Cheng and N. Wu, "Eigen-convergence of Gaussian kernelized graph Laplacian by manifold heat interpolation," *Applied and Computational Harmonic Analysis*, vol. 61, pp. 132–190, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S106352032200046X

[21] Z. Wang, L. Ruiz, and A. Ribeiro, "Convolutional filtering on sampled manifolds," 2022. [Online]. Available: https://arxiv.org/abs/2211.11058

[22] B. Landa and X. Cheng, "Robust inference of manifold density and geometry by doubly stochastic scaling," *arXiv preprint arXiv:2209.08004*, 2022.

[23] D. B. Dunson, H.-T. Wu, and N. Wu, "Spectral convergence of graph laplacian and heat kernel reconstruction in l-infinity from random samples," *Applied and Computational Harmonic Analysis*, vol. 55, pp. 282–336, 2021.

[24] A. Little, D. McKenzie, and J. M. Murphy, "Balancing geometry and density: Path distances on high-dimensional data," *SIAM Journal on Mathematics of Data Science*, vol. 4, no. 1, pp. 72–99, 2022. [Online]. Available: https://doi.org/10.1137/20M1386657