Generalization of Geometric Graph Neural Networks

Zhiyang Wang Juan Cerviño Alejandro Ribeiro

***Abstract*—In this paper, we study the generalization capabilities of graph neural networks (GNNs). We consider the case when the graph is constructed from a finite set of randomly sampled points over an embedded manifold while the sampling can be non-uniform. We construct the graph as relatively sparse which is a reasonable model in many application scenarios. We prove a generalization gap between the optimal empirical risk and optimal statistical risk of this GNN, which increases with the dimension of the underlying manifold and decreases with the number of sampled points from the manifold. This generalization gap ensures that the GNN trained on a graph constructed based on a finite set of sampled points can be utilized to process other unseen graphs constructed from the same underlying manifold. The generalization gap is derived based on the non-asymptotic convergence result of a GNN on the sampled graph to the underlying manifold neural networks (MNNs). We verify this theoretical result with experiments on a citation network.**

***Index Terms*—Graph neural networks, generalization analysis, manifold neural networks, non-asymptotic convergence**

I. Introduction

Graph structures can describe many modern datasets with relationships well captured. Convolutional filters and neural networks on graphs [1]–[4] have become the top choice to process information over graphs. A graph as a discrete model can naturally represent a discrete data structure. Examples include social networks [5], [6], protein structures [7], [8], multi-agent control [9]. In many practical scenarios, graphs can be seen as samples from a manifold, approximating the continuous topological space with finite samples, as in the case of point clouds [10], data manifolds [11], and irregular space navigation [12]. Under this context, the graph convolutional filters and GNNs can be seen as approximations of the counterparts on the manifold [13], [14]. This implies that convolutional structures on graphs with finite sample points can capture the underlying geometry of the manifold.

The main technical contribution of this paper is analyzing the generalization capabilities of GNNs operated on a sampled graph from an underlying manifold. We show that GNNs trained on a graph can be implemented on other unseen graphs sampled from this manifold. We consider this manifold as a common underlying structure for the graphs which helps to derive a generalization bound that decreases with the number of sampled points.

Formally, we are given a set of N i.i.d. points X_N over an embedded manifold $\mathcal{M}$. Input and target graph signals $\mathbf{x}_N$ and $\mathbf{y}_N$ are sampled from underlying manifold signals. The goal is to learn a GNN $\mathbf{\Phi}$ that estimates $\mathbf{y}_N$ with $\mathbf{\Phi}(\mathbf{H}, \mathbf{x}_N)$ where $\mathbf{H} \in \mathcal{H}$ represent the filter parameter set. We use a L^2 loss function ℓ to measure the estimation performance between true target $\mathbf{y}_N$ and the estimated target $\mathbf{\Phi}(\mathbf{x}_N)$. Practically, the GNN is trained to minimize an empirical risk written as

$$R_E(\mathbf{H}) = \ell(\mathbf{\Phi}(\mathbf{H}, \mathbf{x}_N), \mathbf{y}_N). \tag{1}$$

While theoretically, a machine learning algorithm aims to minimize the statistical risk as

$$R_S(\mathbf{H}) = \mathbb{E}_{X_N}\left[\ell\left(\mathbf{\Phi}(\mathbf{H}, \mathbf{x}_N), \mathbf{y}_N\right)\right]. \tag{2}$$

The generalization gap is defined to be

$$GA = \min_{\mathbf{H}\in\mathcal{H}} R_S(\mathbf{H}) - \min_{\mathbf{H}\in\mathcal{H}} R_E(\mathbf{H}). \tag{3}$$

We analyze this generalization gap by establishing the convergence of GNNs to the neural networks built on the underlying manifold, which is manifold neural network [13], [15]. Based on the convergence results, we can prove that the generalization gap between the finite N training points on the sampled graph and the true distribution of these N points is small and decreases with the number of sampled points. The following shows an informal statement of our main Theorem 1.

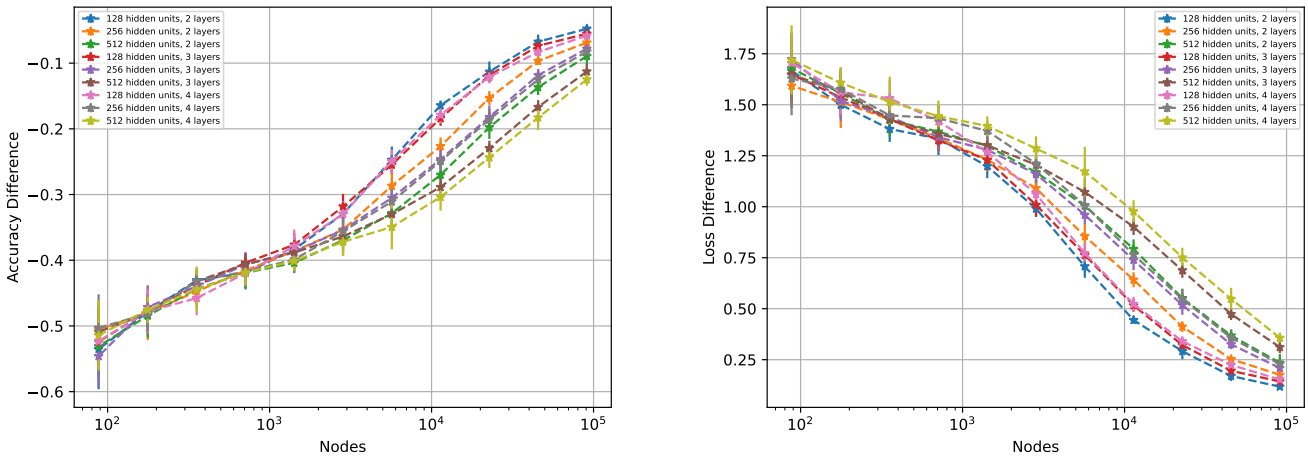
Theorem (Informal)**.** *Consider a graph constructed on N i.i.d. randomly sampled points over a d-dimensional manifold $\mathcal{M}$ with respect to the measure μ over the manifold. Then, the generalization gap of a GNN trained on this graph satisfies with probability $1-\delta$ that*

$$GA = \mathcal{O}\left(\left(\frac{\log N/\delta}{N}\right)^{\frac{1}{d+4}}\right). \tag{4}$$

This shows that the generalization gap decreases approximately polynomially with respect to the number of points N while the exponent is related to the dimension of the underlying manifold. We further attest this theoretical result with a citation network classification problem, where we can observe that the order of the decreasing with respect to the number of sampled points as shown in Figure 1.

In [14], [16]–[19], transferability of GNNs are analyzed by comparing the output difference of GNNs on different sizes of graphs when graphs converge to a limit model as manifold or graphon without generalization analysis. In [20], [21], the authors show how increasing the size of the graph as the GNN learns, generalizes to the large-scale graph. In [22], the authors prove generalization bound of GNNs with VC-dimension which is commonly used for convolutional neural networks. In [23], the authors derive a generlization bound for a single layer GNN based on the stability analysis, with the bound scaling with the largest eigenvalue of the graph Laplacian. In [24], the authors provide a generalization bound based on PAC-

Supported by Theorinet Simons. The authors are with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, 19104, USA.



(a) Accuracy Generalization Gap

(b) Loss Generalization Gap

Fig. 1: OBGN-Arxiv accuracy and loss generalization gaps. The GNN is trained over the number of nodes indicated on the x-axis, and then the generalization gap (difference between the training and loss datasets) is measured, both in terms of the accuracy, and the loss with which we trained – the cross-entropy loss. As can be seen, in both cases the generalization gap presents a linear behavior with respect to the logarithm of the number of nodes.

Bayesian analysis with the bound depending on the maximum degree of the graph and the spectral norms. In [25], the authors provide a generalization bound on message passing networks comparable to Rademacher bounds in recurrent neural networks. The previous generalization gaps scale with the size of graph without capturing the underlying common structures of the graphs. A generalization analysis is carried out in [26], [27], where the graphs are on message-passing neural networks on graphs that are randomly sampled from a collection of template random graph models. We study the limit of graphs as a manifold, which is more realistic in practice and is more suitable to process high-dimensional data.

The rest of the paper is organized as follows. We start with preliminary concepts of graph neural networks (GNNs) and manifold neural networks (MNNs) in Section II. In Section III, we construct the relatively sparse graphs by sampled points from the manifold and we present the generalization gap of GNNs on constructed graphs based on the convergence of GNNs to the underlying MNN. Our proposed results are verified in a citation classification problem in Section IV. The conclusions are presented in Section V.

II. Preliminaries

Let us start with the basic definitions of graph neural networks and manifold neural networks.

A. Graph Convolutions and Graph Neural Networks

An undirected graph $\mathbf{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ contains N nodes with a node set $\mathcal{V}$ and edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The weights of the edges are assigned by $\mathcal{W} : \mathcal{E} \to \mathbb{R}$. Graph signals $\mathbf{x} \in \mathbb{R}^N$ map values to each node. A graph shift operator (GSO) [3], [4] $\mathbf{S} \in \mathbb{R}^{n \times n}$ is a graph matrix with $[\mathbf{S}]_{ij} \neq 0$ if and only if $(i, j) \in \mathcal{E}$ or $i = j$, e.g., the graph Laplacian $\mathbf{L}$. The GSO can shift or diffuse signals to each node by aggregating signal values of neighbors. A graph convolution is defined based on a consecutive graph shift operation. A graph convolutional filter $\mathbf{h_G}$ [1], [2], [4] with filter coefficients $\{h_k\}_{k=0}^{K-1}$ is formally defined as

$$\mathbf{h_G}(\mathbf{S})\mathbf{x} = \sum_{k=0}^{K-1} h_k \mathbf{S}^k \mathbf{x}. \tag{5}$$

Replacing $\mathbf{S}$ with the spectral decomposition in (5), we observe that the spectral representation of a graph filter is

$$\mathbf{V}^H \mathbf{h_G}(\mathbf{S})\mathbf{x} = \sum_{k=1}^{K-1} h_k \mathbf{\Lambda}^k \mathbf{V}^H \mathbf{x} = h(\mathbf{\Lambda})\mathbf{V}^H \mathbf{x}. \tag{6}$$

This leads to a point-wise frequency response of the graph convolution, which is $h(\lambda) = \sum_{k=0}^{K-1} h_k \lambda^k$, depending only on the weights $\{h_k\}_{k=0}^{K-1}$ and on the eigenvalues of $\mathbf{S}$.

A graph neural network (GNN) is composed of cascading layers that each consists of a bank of graph convolutional filters followed by a point-wise nonlinearity $\sigma : \mathbb{R} \to \mathbb{R}$. Specifically, the l-th layer of a GNN that produces F_l output features $\{\mathbf{x}_l^p\}_{p=1}^{F_l}$ with F_{l-1} input features $\{\mathbf{x}_{l-1}^q\}_{q=1}^{F_{l-1}}$ is written as

$$\mathbf{x}_l^p = \sigma\left(\sum_{q=1}^{F_{l-1}} \mathbf{h}_{\mathbf{G}}^{lpq}(\mathbf{S})\mathbf{x}_{l-1}^q\right), \tag{7}$$

for each layer $l = 1, 2 \cdots, L$. The graph filter $\mathbf{h}_{\mathbf{G}}^{lpq}(\mathbf{S})$ maps the q-th feature of layer $l-1$ to the p-th feature of layer l as (5). We denote the GNN like (7) as a mapping $\mathbf{\Phi_G}(\mathbf{H}, \mathbf{S}, \mathbf{x})$ for the ease of presentation, where $\mathbf{H} \in \mathcal{H} \subset \mathbb{R}^P$ denotes a set of the graph filter coefficients at all layers and $\mathcal{H}$ denotes the

set of all possible parameter sets.

B. Manifold Convolutions and Manifold Neural Networks

We consider a d-dimensional compact, smooth and differentiable submanifold $\mathcal{M}$ embedded in $\mathbb{R}^{\mathsf{N}}$. The embedding induces a probability measure μ over the manifold with density function $\rho : \mathcal{M} \to (0, \infty)$, which is assumed to be bounded as $0 < \rho_{min} \leq \rho \leq \rho_{max} < \infty$. Manifold signals [15] are likewise defined as scalar functions $f : \mathcal{M} \to \mathbb{R}$. We use $L^2(\mathcal{M})$ to denote L^2 functions over $\mathcal{M}$ with respect to measure μ. The inner product of signals $f, g \in L^2(\mathcal{M})$ is defined as

$$\langle f, g \rangle_{\mathcal{M}} = \int_{\mathcal{M}} f(x) g(x) \mathrm{d}\mu(x), \tag{8}$$

with the L^2 norm defined as $\|f\|^2_{\mathcal{M}} = \langle f, f \rangle_{\mathcal{M}}$. The manifold with density is endowed with a weighted Laplace operator [28], which generalizes the Laplace-Beltrami operator as

$$\mathcal{L}_\rho f = -\frac{1}{2\rho} \mathrm{div}(\rho^2 \nabla f), \tag{9}$$

with div the divergence operator of $\mathcal{M}$ and ∇ the gradient operator of $\mathcal{M}$ [10], [29]. Manifold shift operation is defined relying on the Laplace operator $\mathcal{L}_\rho$ and on the heat diffusion process over the manifold (see [15] for a detailed exposition). For a manifold signal $f \in L^2(\mathcal{M})$, the manifold shift can be explicitly written as $e^{-\mathcal{L}_\rho} f$. Analogous to graph convolution, manifold convolution [15] can be defined in a shift-and-sum manner as

$$g(x) = \mathbf{h}(\mathcal{L}_\rho) f(x) = \sum_{k=0}^{K-1} h_k e^{-k\mathcal{L}_\rho} f(x). \tag{10}$$

Consider the Laplace operator is self-adjoint and positive-semidefinite and the manifold $\mathcal{M}$ is compact, $\mathcal{L}_\rho$ has real, positive and discrete eigenvalues $\{\lambda_i\}_{i=1}^{\infty}$, written as $\mathcal{L}_\rho \boldsymbol{\phi}_i = \lambda_i \boldsymbol{\phi}_i$ where $\boldsymbol{\phi}_i$ is the eigenfunction associated with eigenvalue λ_i. The eigenvalues are ordered in increasing order as $0 < \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \ldots$, and the eigenfunctions are orthonormal and form an eigenbasis of $L^2(\mathcal{M})$. When mapping a manifold signal onto the eigenbasis $[\hat{f}]_i = \langle f, \boldsymbol{\phi}_i \rangle_{L^2(\mathcal{M})} = \int_{\mathcal{M}} f(x) \boldsymbol{\phi}_i(x) \mathrm{d}\mu(x)$, the manifold convolution can be written in the spectral domain as

$$[\hat{g}]_i = \sum_{k=0}^{K-1} h_k e^{-k\lambda_i} [\hat{f}]_i. \tag{11}$$

Hence, the frequency response of manifold filter is given by $\hat{h}(\lambda) = \sum_{k=0}^{K-1} h_k e^{-k\lambda}$, depending only on the filter coefficients h_k and eigenvalues of $\mathcal{L}_\rho$.

Manifold neural network (MNN) is built by cascading L layers, each of which consists of a bank of manifold filters and a pointwise nonlinearity σ. Each layer $l = 1, 2 \cdots, L$ can be explicitly denoted as

$$f_l^p(x) = \sigma \left(\sum_{q=1}^{F_{l-1}} \mathbf{h}_l^{pq}(\mathcal{L}_\rho) f_{l-1}^q(x) \right), \tag{12}$$

where f_{l-1}^q is an input feature and f_l^p, $1 \leq p \leq F_l$ is a an output feature. In each layer manifold filters maps F_{l-1} input features to F_l output features. To represent the MNN succinctly, we group all learnable parametes, and we denote the mapping as $\boldsymbol{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f)$, where $\mathbf{H} \in \mathcal{H} \subset \mathbb{R}^P$ is a filter parameter set of the manifold filters.

III. Convergence and Generalization Analysis of GNNs via MNNs

Suppose we are given an embedded manifold $\mathcal{M} \subset \mathbb{R}^{\mathsf{N}}$ with input manifold signal $f \in L^2(\mathcal{M})$ and target manifold signal $g \in L^2(\mathcal{M})$ attached to it. An MNN, as defined in (12), predicts the target signal with $\boldsymbol{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f)$ where $\mathbf{H} \in \mathcal{H} \subset \mathbb{R}^P$ is the set of filter coefficients, $\mathcal{L}_\rho$ is the weighted Laplacian defined in (9) and f the input manifold signal. A positive loss function is denoted as $\ell(\boldsymbol{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f), g)$ to measure the estimation performance.

Practically, it is normal to access the underlying topological space by sampling discrete points over the continuous domain. Suppose we are given pairs of graph signals and graphs $(\mathbf{x}_N, \mathbf{G}_N)$ along with target output graph signals $\mathbf{y}_N \in \mathbb{R}^N$. Graph $\mathbf{G}_N$ is constructed based on a set of N i.i.d. randomly sampled points $X_N = \{x_N^1, x_N^2, \cdots, x_N^N\}$ according to measure μ over the underlying manifold $\mathcal{M}$. These N sampled points are seen as nodes. Further, every pair of nodes (x_N^i, x_N^j) is connected by an edge with weight value $[\mathbf{W}_N]_{ij}, \mathbf{W}_N \in \mathbb{R}^{N \times N}$ determined by a function K_ϵ of their Euclidean distance [30], which is explicitly written as

$$[\mathbf{W}_m]_{ij} = K_\epsilon(x_N^i, x_N^j) = \frac{\alpha_d}{(d+2) N \epsilon^{d+2}} \mathbb{1} \left(\frac{|x_N^i - x_N^j|}{\epsilon} \right), \tag{13}$$

where α_d is the volume of the d-dimensional Euclidean unit ball and $\mathbb{1}$ stands for a characteristic function. Graph input and target output signals $\mathbf{x}_N, \mathbf{y}_N \in \mathbb{R}^N$ are supported on sampled points X_N belonging to $L^2(X_N)$, whose values are sampled from manifold input signal f and target signal g respectively, written explicitly as

$$[\mathbf{x}_N]_i = f(x_N^i), \quad [\mathbf{y}_N]_i = g(x_N^i). \tag{14}$$

We define a sampling operator $\mathbf{P}_N : L^2(\mathcal{M}) \to L^2(X_N)$ to represent this mapping. As introduced in Sec. II, we denote a GNN mapping as $\boldsymbol{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{x}_N)$, where $\mathbf{H} \in \mathcal{H} \subset \mathbb{R}^P$ is the set of filter coefficients, $\mathbf{L}_N = \mathrm{diag}(\mathbf{W}_N \mathbf{1}) - \mathbf{W}_N$ as the graph Laplacian which is implemented as a GSO, $\mathbf{x}_N$ as the input graph signal. A positive loss function is denoted as $\ell(\boldsymbol{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{x}_N), \mathbf{y}_N)$ to measure the estimation performance.

A. Convergence of GNNs to MNNs

We first show that the difference between the outputs of MNN $\boldsymbol{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f)$ and the GNN on the sampled graph with N nodes $\boldsymbol{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{x}_N)$ can be bounded, with the bound decreasing with the number of the nodes sampled from the manifold.

Suppose the input manifold signal f is λ_M bandlimited, which is explicitly defined as follows.

Definition 1. *A manifold signal f is λ_M bandlimited if for all eigenpairs $\{\lambda_i, \boldsymbol{\phi}_i\}_{i=1}^{\infty}$ of the weighted Laplacian $\mathcal{L}_\rho$ when $\lambda_i > \lambda_M$, we have $\langle f, \boldsymbol{\phi}_i \rangle_{\mathcal{M}} = 0$.*

We denote M as the cardinality of the limited spectrum of $\mathcal{L}_\rho$, i.e. $M = \#\{\lambda_i < \lambda_M\}$. We also need to put an assumption on the frequency response function of the filters as follows.

Definition 2. *A filter is a low pass filter if its frequency response satisfies*

$$\left|\hat{h}(a)\right| = \mathcal{O}\left(a^{-d}\right), \quad a \in (0, \infty). \tag{15}$$

For the nonlinearity functions utilized in the GNNs, we need to make the following assumption.

Assumption 1. *(Normalized Lipschitz nonlinearity) The nonlinearity σ is normalized Lipschitz continuous, i.e., $|\sigma(a)-\sigma(b)| \leq |a-b|$, with $\sigma(0) = 0$.*

We note that this assumption is reasonable considering most common activation functions are normalized Lipschitz, such as ReLU, modulus and sigmoid.

With the low-pass filters and normalized Lipschitz nonlinearities in both the GNN and MNN, we are ready to prove a difference bound between the outputs of a GNN operating on a sampled graph over $\mathcal{M}$ and the outputs of an MNN on $\mathcal{M}$.

Proposition 1. *Let $\mathcal{M} \subset \mathbb{R}^{\mathsf{N}}$ be an embedded manifold with weighted Laplace operator $\mathcal{L}_\rho$ and a λ_M-bandlimited manifold signal f. Consider a pair of graph and graph signal $(\mathbf{x}_N, \mathbf{G}_N)$ with N nodes sampled i.i.d. with measure μ over $\mathcal{M}$. The graph Laplacian $\mathbf{L}_N$ is calculated based on* (13)*. Let $\mathbf{\Phi}(\mathbf{H}, \mathcal{L}_\rho, \cdot)$ be a single layer MNN on $\mathcal{M}$* (12) *with single input and output features. Let $\mathbf{\Phi}(\mathbf{H}, \mathbf{L}_N, \cdot)$ be the GNN with the same architecture applied to the graph $\mathbf{G}_N$. Then, with the filters as low-pass and nonlinearities as normalized Lipschitz continuous, it holds in probability at least $1-\delta$ that*

$$\begin{aligned}\|\mathbf{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{P}_N f) - \mathbf{P}_N \mathbf{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f)\|_2 \leq \\ C_1 \left(\frac{\log \frac{C_1 N}{\delta}}{N}\right)^{\frac{1}{d+4}} + C_2 \left(\frac{\log \frac{C_1 N}{\delta}}{N}\right)^{\frac{1}{d+4}} \theta_M^{-1} \\ + C_3 \sqrt{\frac{\log(1/\delta)}{N}} + C_4 M^{-1},\end{aligned} \tag{16}$$

where C_1, C_2, C_3 and C_4 are constants defined in Appendix A and $\theta_M = \min_{i=1,2\cdots,M} |\lambda_i - \lambda_{i+1}|$.

Corollary 1. *The difference of the L-layer GNN $\mathbf{\Phi}(\mathbf{H}, \mathbf{L}_N, \cdot)$ on a graph $\mathbf{G}_N$ sampled from the manifold and MNN $\mathbf{\Phi}(\mathbf{H}, \mathcal{L}_\rho, \cdot)$ converges to zero as N goes to infinity.*

Proof. We denote the four terms in (16) as $A_1(N)$, $A_2(M, N)$, $A_3(N)$ and $A_4(M)$. For every $\delta > 0$, we can choose some M_0 such at $A_4(M_0) < \delta/2$. There exists some n_0 such that for all $N > n_0$, $A_1(N) + A_2(M_0, N) + A_3(N) < \delta/2$. Therefore, this satisfies the definition of the convergence, which implies that for every $\delta > 0$, there exists some n_0 so that for all $N > n_0$, we have $A_1(N) + A_2(M_0, N) + A_3(N) + A_4(M_0) < \delta$. □

Proposition 1 shows that the output difference of GNN on the sampled graph and the underlying MNN is bounded in high probability with the bound decreasing with the number of sampled points, i.e. the number of nodes in the graph. Even as the spectrum becomes larger, there always exists some N large enough to make the difference goes to zero as N increases. This attest the convergence of GNN on the sampled graph to MNN. With this bound holding in high probability, we can naturally derive the difference bound in expectation of N randomly sampled points as follows.

Corollary 2. *The difference bound between GNN and MNN also holds in expectation since each node in X_N is sampled i.i.d. according to measure μ over $\mathcal{M}$*

$$\begin{aligned}\mathop{\mathbb{E}}_{X_N \sim \mu^N} [\|\mathbf{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{P}_N f) - \mathbf{P}_N \mathbf{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f)\|_2] \leq \\ C' N^{-\frac{1}{d+4}} + C'' N^{-\frac{1}{2}} + C''' \left(\frac{\log N}{N}\right)^{\frac{1}{d+4}} + \bar{M} e^{-N/C} \sqrt{N},\end{aligned} \tag{17}$$

where C', C'', C''' and $\bar{M}$ are specified in Appendix B.

We can observe that the output difference decreases with the number of nodes sampled from the underlying manifold, while increasing with the manifold dimension, i.e. the model complexity.

B. Generalization of GNNs

Suppose a GNN is trained over the given graph and graph signals $\mathbf{x}_N, \mathbf{G}_N$. The training intends to minimize the empirical risk defined as

$$P_E^* = \min_{\mathbf{H} \in \mathcal{H}} R_E(\mathbf{H}) := \ell(\mathbf{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{x}_N), \mathbf{y}_N). \tag{18}$$

Given the fact that we only have access to a training set with limited training samples and not the underlying probability distribution μ, nor the manifold $\mathcal{M}$, we can only aim to minimize the empirical risk in practice. Substantially, the goal of the GNN is to minimize the statistical risk, i.e.

$$P_S^* = \min_{\mathbf{H} \in \mathcal{H}} R_S(\mathbf{H}) := \mathbb{E}_{X_N} \left[\ell\left(\mathbf{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{x}_N), \mathbf{y}_N\right)\right]. \tag{19}$$

where the expectation is taken with respect to N randomly sampled points $X_N \sim \mu^N$.

The generalization gap of GNN is defined to be

$$GA = P_S^* - P_E^*, \tag{20}$$

which measures the difference between the optimal empirical risk and the optimal statistical risk of the GNN. Based on the convergence results that we have derived, we can bound the generalization gap as the following theorem.

Theorem 1. *The Generalization Gap of GNN trained on $(\mathbf{x}_N, \mathbf{G}_N)$ is bounded in probability at least $1-\delta$ that,*

$$GA = \mathcal{O}\left(\left(\frac{\log \frac{N}{\delta}}{N}\right)^{\frac{1}{d+4}} + \left(\frac{\log N}{N}\right)^{\frac{1}{d+4}}\right). \tag{21}$$

We observe that the generalization gap decreases with the number of sampled points over the manifold N. The restriction put on the graphs that they are sampled from the same underlying manifold makes the GNN generalize better without depending on the VC dimension and Rademacher

Layers	Hidden Units	Pearson Correlation	p-value
1	64	-0.50140	0.11612
1	128	-0.50263	0.11508
1	256	-0.50932	0.10955
1	512	-0.49895	0.11820
2	64	-0.69724	0.01709
2	128	-0.75542	0.00718
2	256	-0.80591	0.00274
2	512	-0.83898	0.00124
3	64	-0.70005	0.01646
3	128	-0.76771	0.00580
3	256	-0.82561	0.00175
3	512	-0.86436	0.00060
4	64	-0.69280	0.01811
4	128	-0.76440	0.00615
4	256	-0.84140	0.00117
4	512	-0.87543	0.00041

TABLE I: Pearson Correlation index and associated p-value for generalization error over the cross-entropy loss.

Layers	Hidden Units	Pearson Correlation	p-value
1	64	0.52974	0.09374
1	128	0.53093	0.09287
1	256	0.54261	0.08459
1	512	0.52894	0.09433
2	64	0.71715	0.01299
2	128	0.77199	0.00537
2	256	0.82032	0.00198
2	512	0.84979	0.00093
3	64	0.71547	0.01331
3	128	0.77802	0.00481
3	256	0.82619	0.00172
3	512	0.86104	0.00066
4	64	0.70317	0.01578
4	128	0.77362	0.00521
4	256	0.84437	0.00108
4	512	0.87329	0.00045

TABLE II: Pearson Correlation index and associated p-value for generalization error over the accuracy.

complexity. Furthermore, the generalization gap increases with the dimension d, which represents the complexity of the underlying manifold. That is to say, for a high-complexity manifold model, we can sample more points over this manifold to build a graph, on which the GNN can achieve a small generalization gap.

IV. Simulations

We consider an experiment to showcase the generalization capabilities of Graph Neural Networks. To do so, we trained a GNN on a subset of nodes, and measured the generalization gap as a function of the number of nodes in the graph. We conducted experiments in a real-world dataset called OGBN-Arxiv [31]. OGBN-Arxiv graph has $169,343$ nodes and $1,166,243$ edges and it represents the citation network between computer science arXiv papers. Each node is a paper, and the graph signals at each node are 128 dimensional embeddings of the title and abstract of each paper [32]. The graph labels $\mathbf{y}_i$ are one of the 40 categories the paper belongs to.

We trained a GNN using the graph convolution layer, and relu as the non-linearity. We used trained using $\{1, 2, 3, 4\}$ layers, and $\{64, 128, 256, 512\}$ hidden units. In all cases, we run each experiment for 1000 epochs, using 10 different seeds. On the optimization side, we trained using a learning rate of 0.005. To train we used the cross-entropy loss.

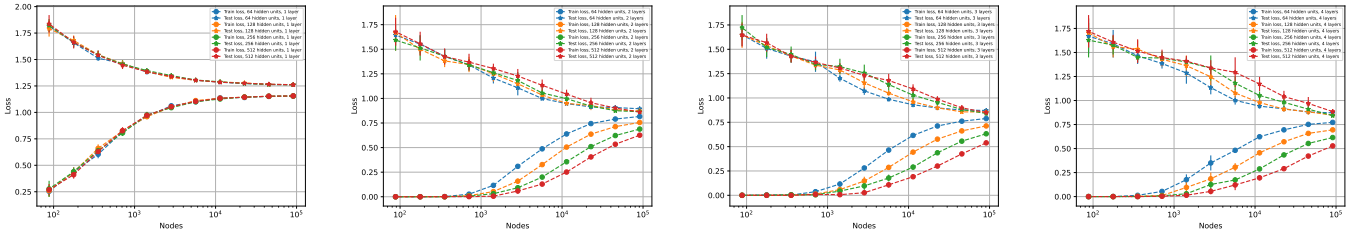
As can be seen in Figure 3d, both in terms of the cross-entropy loss 1b, as well as the accuracy 1a, the generalization gap shows a linear behavior with the logarithm of the number of nodes in the training graph. This finding can be formalized by looking at the Pearson correlation index in Tables II, and I. The Pearson correlation index is a number between $\breve{}1$ and 1 that measures the linearity of two variables, in this case the logarithm of the number of nodes, and the generalization gap. The closer the magnitude of the Pearson correlation index is to 1, the stronger the linear relationship is. For GNNs with more than one layer, the Pearson correlation index is above 0.7, showing a strong linear correlation. This validates the claims that we put forward.

We can also look at the loss and accuracy as a number of features in the convolution. In Figure 4, we plot the training and testing losses, and accuracies as a function of the number of layers, and inner features of the GNN. There are two clear patterns, the one layer GNN and more than one layer. In the case of one layer, the GNN is unable to overfit the training set, whereas in the GNNs with more convolutional layers, the GNN is able to overfit the training set with a small number of nodes on it. The second salient difference between these two is the learning behavior when the width of the GNN increases. With one convolutional layer, the width is almost irrelevant, which is not the case in the other 3 experiments. This shows that one layer GNNs are unable to learn properly, even when the width of the GNN increases. This is not the case in $\{2, 3, 4\}$ layer GNNs. In this case, as the number of hidden units increases, so does the training accuracy. That is to say, if the number of hidden units is larger, the accuracy drop as the number of nodes in the training set increases is slower. However, more hidden units, does not necessarily mean better accuracy in the test set.

In all, we validated the claim that the generalization gap increases linearly with the logarithm of the number of nodes in the training graph.

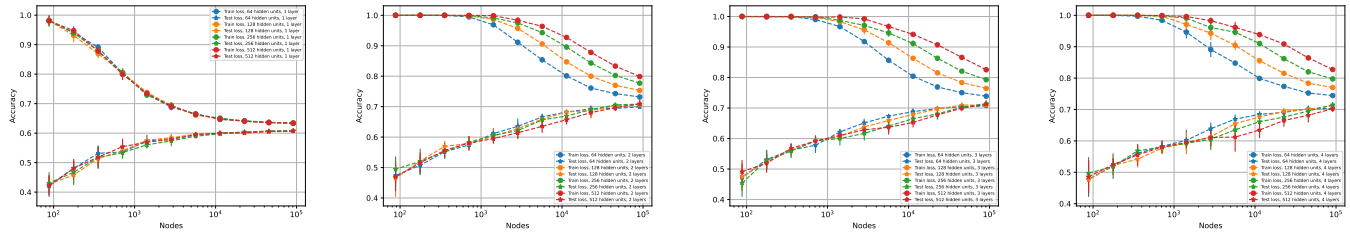
V. Conclusion

In this paper, we implement the manifold convolutional neural networks as a limit model for graph neural networks when graphs are sampled from this manifold. With graph constructed with the i.i.d. randomly sampled points over the manifold, we can prove the GNN converges to the MNN with a non-asymptotic rate both in probability and in expectation. Based



(a) One Layer (b) Two Layers (c) Three Layers (d) Four Layers

Fig. 3: OBGN-Arxiv training and testing losses for $\{1, 2, 3, 4\}$ layers.



(a) One Layer (b) Two Layers (c) Three Layers (d) Four Layers

Fig. 4: OBGN-Arxiv training and testing accuracies for $\{1, 2, 3, 4\}$ layers.

on these convergence results, we prove the generalization gap of the GNN over finite training samples and the real distribution. We show that the generalization gap decreases with the number of sampled points while increases with the manifold dimension. We finally verify our convergence and generalization results numerically with a citation classification problem.

REFERENCES

[1] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs," *IEEE transactions on signal processing*, vol. 61, no. 7, pp. 1644–1656, 2013.

[2] F. Gama, A. G. Marques, G. Leus, and A. Ribeiro, "Convolutional neural network architectures for signals supported on graphs," *IEEE Transactions on Signal Processing*, vol. 67, no. 4, pp. 1034–1049, 2019.

[3] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE signal processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[4] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[5] C. Huang, H. Xu, Y. Xu, P. Dai, L. Xia, M. Lu, L. Bo, H. Xing, X. Lai, and Y. Ye, "Knowledge-aware coupled graph neural network for social recommendation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 4115–4122, 2021.

[6] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, "Graph neural networks in recommender systems: a survey," *ACM Computing Surveys*, vol. 55, no. 5, pp. 1–37, 2022.

[7] A. Strokach, D. Becerra, C. Corbi-Verge, A. Perez-Riba, and P. M. Kim, "Fast and flexible protein design using deep graph neural networks," *Cell systems*, vol. 11, no. 4, pp. 402–411, 2020.

[8] N. Yin, L. Shen, M. Wang, L. Lan, Z. Ma, C. Chen, X.-S. Hua, and X. Luo, "Coco: A coupled contrastive framework for unsupervised domain adaptive graph classification," in *International Conference on Machine Learning*, pp. 40040–40053, PMLR, 2023.

[9] W. Gosrich, S. Mayya, R. Li, J. Paulos, M. Yim, A. Ribeiro, and V. Kumar, "Coverage control in multi-robot systems via graph neural networks," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 8787–8793, IEEE, 2022.

[10] 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.

[11] U. Sharma and J. Kaplan, "Scaling laws from the data manifold dimension," *Journal of Machine Learning Research*, vol. 23, no. 9, pp. 1–34, 2022.

[12] J. Cervino, L. F. O. Chamon, B. D. Haeffele, R. Vidal, and A. Ribeiro, "Learning globally smooth functions on manifolds," in *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, eds.), vol. 202 of *Proceedings of Machine Learning Research*, pp. 3815–3854, PMLR, 23–29 Jul 2023.

[13] Z. Wang, L. Ruiz, and A. Ribeiro, "Convolutional neural networks on manifolds: From graphs and back," in *2022 56th Asilomar Conference on Signals, Systems, and Computers*, pp. 356–360, IEEE, 2022.

[14] Z. Wang, L. Ruiz, and A. Ribeiro, "Geometric graph filters and neural networks: Limit properties and discriminability trade-offs," *arXiv preprint arXiv:2305.18467*, 2023.

[15] Z. Wang, L. Ruiz, and A. Ribeiro, "Stability to deformations of manifold filters and manifold neural networks," *IEEE Transactions on Signal Processing*, pp. 1–15, 2024.

[16] L. Ruiz, L. Chamon, and A. Ribeiro, "Graphon neural networks and the transferability of graph neural networks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1702–1712, 2020.

[17] L. Ruiz, L. F. Chamon, and A. Ribeiro, "Transferability properties of graph neural networks," *IEEE Transactions on Signal Processing*, 2023.

[18] R. Levie, W. Huang, L. Bucci, M. Bronstein, and G. Kutyniok, "Transferability of spectral graph convolutional neural networks," *Journal of Machine Learning Research*, vol. 22, no. 272, pp. 1–59, 2021.

[19] S. Maskey, R. Levie, and G. Kutyniok, "Transferability of graph neural networks: an extended graphon approach," *Applied and Computational Harmonic Analysis*, vol. 63, pp. 48–83, 2023.

[20] J. Cerviño, L. Ruiz, and A. Ribeiro, "Learning by transference: Training graph neural networks on growing graphs," *IEEE Transactions on Signal Processing*, vol. 71, pp. 233–247, 2023.

[21] J. Cerviño, L. Ruiz, and A. Ribeiro, "Training graph neural networks on growing stochastic graphs," in *ICASSP 2023 - 2023 IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1–5, 2023.

[22] F. Scarselli, A. C. Tsoi, and M. Hagenbuchner, "The vapnik–chervonenkis dimension of graph and recursive neural networks," *Neural Networks*, vol. 108, pp. 248–259, 2018.

[23] S. Verma and Z.-L. Zhang, "Stability and generalization of graph convolutional neural networks," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1539–1548, 2019.

[24] R. Liao, R. Urtasun, and R. Zemel, "A pac-bayesian approach to generalization bounds for graph neural networks," in *International Conference on Learning Representations*, 2020.

[25] V. Garg, S. Jegelka, and T. Jaakkola, "Generalization and representational limits of graph neural networks," in *International Conference on Machine Learning*, pp. 3419–3430, PMLR, 2020.

[26] S. Maskey, R. Levie, Y. Lee, and G. Kutyniok, "Generalization analysis of message passing neural networks on large random graphs," *Advances in neural information processing systems*, vol. 35, pp. 4805–4817, 2022.

[27] S. Maskey, G. Kutyniok, and R. Levie, "Generalization bounds for message passing networks on mixture of graphons," *arXiv preprint arXiv:2404.03473*, 2024.

[28] A. Grigor'yan, "Heat kernels on weighted manifolds and applications," *Cont. Math*, vol. 398, no. 2006, pp. 93–191, 2006.

[29] G. Gross and E. Meinrenken, *Manifolds, vector fields, and differential forms: an introduction to differential geometry*. Springer Nature, 2023.

[30] J. Calder and N. G. Trillos, "Improved spectral convergence rates for graph laplacians on ε-graphs and k-nn graphs," *Applied and Computational Harmonic Analysis*, vol. 60, pp. 123–175, 2022.

[31] K. Wang, Z. Shen, C. Huang, C.-H. Wu, Y. Dong, and A. Kanakia, "Microsoft academic graph: When experts are not enough," *Quantitative Science Studies*, vol. 1, no. 1, pp. 396–413, 2020.

[32] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[33] U. Von Luxburg, M. Belkin, and O. Bousquet, "Consistency of spectral clustering," *The Annals of Statistics*, pp. 555–586, 2008.

[34] W. Arendt, R. Nittka, W. Peter, and F. Steiner, "Weyl's law: Spectral properties of the laplacian in mathematics and physics," *Mathematical analysis of evolution, information, and complexity*, pp. 1–71, 2009.

APPENDIX

A. Proof of Proposition 1

Proposition 2. *[30, Theorem 2.4, Theorem 2.6] Let $\mathcal{M} \subset \mathbb{R}^{\mathsf{N}}$ be equipped with LB operator $\mathcal{L}$, whose eigendecomposition is given by $\{\lambda_i, \boldsymbol{\phi}_i\}_{i=1}^{\infty}$. Let $\mathbf{L}_N$ be the discrete graph Laplacian of graph weights defined as* (13)*, with spectrum $\{\lambda_{i,N}, \boldsymbol{\phi}_{i,N}\}_{i=1}^{N}$. Fix $K \in \mathbb{N}^+$ and assume that $\epsilon = \epsilon(N) \geq (\log(CN/\delta)/N)^{1/(d+4)}$ Then, with probability at least $1-\delta$, we have*

$$|\lambda_i - \lambda_{i,N}| \leq C_{\mathcal{M},1}\lambda_i\epsilon, \quad \|a_i\boldsymbol{\phi}_{i,N} - \boldsymbol{\phi}_i\| \leq C_{\mathcal{M},2}\frac{\lambda_i}{\theta_i}\epsilon, \quad (22)$$

with $a_i \in \{-1, 1\}$ for all $i < K$ and θ the eigengap of $\mathcal{L}$, i.e., $\theta_i = \min\{\lambda_i - \lambda_{i-1}, \lambda_{i+1} - \lambda_i\}$. The constants $C_{\mathcal{M},1}$, $C_{\mathcal{M},2}$ depend on d and the volume of $\mathcal{M}$.

The inner product of signals $f, g \in L^2(\mathcal{M})$ is defined as

$$\langle f, g\rangle_{\mathcal{M}} = \int_{\mathcal{M}} f(x)g(x)\mathrm{d}\mu(x), \quad (23)$$

where $\mathrm{d}\mu(x)$ is the volume element with respect to the measure μ over $\mathcal{M}$. Similarly, the norm of the manifold signal f is

$$\|f\|_{\mathcal{M}}^2 = \langle f, f\rangle_{\mathcal{M}}. \quad (24)$$

Because $\{x_1, x_2, \cdots, x_N\}$ is a set of randomly sampled points from $\mathcal{M}$, based on Theorem 19 in [33] we can claim that

$$|\langle \mathbf{P}_N f, \boldsymbol{\phi}_i\rangle - \langle f, \boldsymbol{\phi}_i\rangle_{\mathcal{M}}| = O\left(\sqrt{\frac{\log(1/\delta)}{N}}\right). \quad (25)$$

This also indicates that

$$\left|\|\mathbf{P}_N f\|^2 - \|f\|_{\mathcal{M}}^2\right| = O\left(\sqrt{\frac{\log(1/\delta)}{N}}\right), \quad (26)$$

which indicates $\|\mathbf{P}_N f\| = \|f\|_{\mathcal{M}} + O((\log(1/\delta)/N)^{1/4})$. We first write out the filter representation as

$$\|\mathbf{h}(\mathbf{L}_N)\mathbf{P}_N f - \mathbf{P}_N\mathbf{h}(\mathcal{L}_\rho)f\| \leq \left\|\sum_{i=1}^{N}\hat{h}(\lambda_{i,N})\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N} - \sum_{i=1}^{M}\hat{h}(\lambda_i)\langle f, \boldsymbol{\phi}_i\rangle_{\mathcal{M}}\mathbf{P}_N\boldsymbol{\phi}_i\right\| \quad (27)$$

$$\leq \left\|\sum_{i=1}^{M}\hat{h}(\lambda_{i,N})\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N} - \sum_{i=1}^{M}\hat{h}(\lambda_i)\langle \mathbf{P}_N f, \boldsymbol{\phi}_i\rangle_{\mathcal{M}}\boldsymbol{\phi}_i + \sum_{i=M+1}^{N}\hat{h}(\lambda_{i,N})\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N}\right\| \quad (28)$$

$$\leq \left\|\sum_{i=1}^{M}\hat{h}(\lambda_{i,N})\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N} - \sum_{i=1}^{M}\hat{h}(\lambda_i)\langle \mathbf{P}_N f, \boldsymbol{\phi}_i\rangle_{\mathcal{M}}\boldsymbol{\phi}_i\right\| + \left\|\sum_{i=M+1}^{N}\hat{h}(\lambda_{i,N})\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N}\right\| \quad (29)$$

The first part of (29) can be decomposed as

$$\left\|\sum_{i=1}^{M}\hat{h}(\lambda_{i,N})\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N} - \sum_{i=1}^{M}\hat{h}(\lambda_i)\langle f, \boldsymbol{\phi}_i\rangle_{\mathcal{M}}\mathbf{P}_N\boldsymbol{\phi}_i\right\| \leq \left\|\sum_{i=1}^{M}\left(\hat{h}(\lambda_{i,N}) - \hat{h}(\lambda_i)\right)\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N}\right\| + \left\|\sum_{i=1}^{M}\hat{h}(\lambda_i)\left(\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N} - \langle f, \boldsymbol{\phi}_i\rangle_{\mathcal{M}}\mathbf{P}_N\boldsymbol{\phi}_i\right)\right\|. \quad (30)$$

In equation (30), the first part relies on the difference of eigenvalues and the second part depends on the eigenvector difference. The square of the first term in (30) is bounded as

$$\left\|\sum_{i=1}^{M}(\hat{h}(\lambda_{i,n}) - \hat{h}(\lambda_i))\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle\boldsymbol{\phi}_{i,N}\right\| \leq \sum_{i=1}^{M}\left|\hat{h}(\lambda_{i,N}) - \hat{h}(\lambda_i)\right||\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N}\rangle| \quad (31)$$

$$\leq \|\mathbf{P}_N f\|\sum_{i=1}^{M}C_{\mathcal{M},1}\epsilon\lambda_i^{-d} \leq \|\mathbf{P}_N f\|C_{\mathcal{M},1}\epsilon\sum_{i=1}^{M}i^{-2} \quad (32)$$

$$\leq \left(\|f\|_{\mathcal{M}} + \left(\frac{\log(1/\delta)}{N}\right)^{\frac{1}{4}}\right)C_{\mathcal{M},1}\epsilon\frac{\pi^2}{6} := A_1(N) \quad (33)$$

In (32), we implement Weyl's law [34] which indicates that

eigenvalues of Laplace operator scales with the order of $i^{2/d}$. The last inequality comes from the fact that $\sum_{i=1}^{\infty} i^{-2} = \frac{\pi^2}{6}$. The second term in (30) can be bounded combined with the convergence of eigenfunctions as

$$\begin{aligned}
&\left\| \sum_{i=1}^{M} \hat{h}(\lambda_i) \left(\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \boldsymbol{\phi}_{i,N} - \langle f, \boldsymbol{\phi}_i \rangle_{\mathcal{M}} \mathbf{P}_N \boldsymbol{\phi}_i \right) \right\| \\
&\leq \left\| \sum_{i=1}^{M} \hat{h}(\lambda_i) \left(\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \boldsymbol{\phi}_{i,N} - \langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \mathbf{P}_N \boldsymbol{\phi}_i \right) \right\| \\
&+ \left\| \sum_{i=1}^{M} \hat{h}(\lambda_i) \left(\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \mathbf{P}_N \boldsymbol{\phi}_i - \langle f, \boldsymbol{\phi}_i \rangle_{\mathcal{M}} \mathbf{P}_N \boldsymbol{\phi}_i \right) \right\| \quad (34)
\end{aligned}$$

The first term in (34) can be bounded as

$$\begin{aligned}
&\left\| \sum_{i=1}^{M} \hat{h}(\lambda_i) \left(\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \boldsymbol{\phi}_{i,N} - \langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle_{\mathcal{M}} \mathbf{P}_N \boldsymbol{\phi}_i \right) \right\| \\
&\leq \sum_{i=1}^{M} \left| \hat{h}(\lambda_i) \right| \|\mathbf{P}_N f\| \|\boldsymbol{\phi}_{i,N} - \mathbf{P}_N \boldsymbol{\phi}_i\| \quad (35) \\
&\leq \sum_{i=1}^{M} (\lambda_i^{-d}) \frac{C_{\mathcal{M},2} \epsilon}{\theta_i} \left(\|f\|_{\mathcal{M}} + \left(\frac{\log(1/\delta)}{N} \right)^{\frac{1}{4}} \right) \quad (36) \\
&\leq C_{\mathcal{M},2} \epsilon \frac{\pi^2}{6} \max_{i=1,\cdots,M} \theta_i^{-1} \left(\|f\|_{\mathcal{M}} + \left(\frac{\log(1/\delta)}{N} \right)^{\frac{1}{4}} \right) \quad (37) \\
&:= A_2(M, N). \quad (38)
\end{aligned}$$

The last equation comes from the definition of norm in $L^2(X_N)$. The second term in (34) can be written as

$$\begin{aligned}
&\left\| \sum_{i=1}^{M} \hat{h}(\lambda_{i,N}) (\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \mathbf{P}_N \boldsymbol{\phi}_i - \langle f, \boldsymbol{\phi}_i \rangle_{\mathcal{M}} \mathbf{P}_N \boldsymbol{\phi}_i) \right\| \\
&\leq \sum_{i=1}^{M} \left| \hat{h}(\lambda_{i,N}) \right| |\langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle - \langle f, \boldsymbol{\phi}_i \rangle_{\mathcal{M}}| \, \|\mathbf{P}_N \boldsymbol{\phi}_i\| \quad (39) \\
&\leq \sum_{i=1}^{M} (\lambda_i^{-d}) \sqrt{\frac{\log(1/\delta)}{N}} \left(1 + \left(\frac{\log(1/\delta)}{N} \right)^{\frac{1}{4}} \right) \quad (40) \\
&\leq \frac{\pi^2}{6} \sqrt{\frac{\log(1/\delta)}{N}} \left(1 + \left(\frac{\log(1/\delta)}{N} \right)^{\frac{1}{4}} \right) := A_3(N) \quad (41)
\end{aligned}$$

The second term in (29) can be bounded as

$$\begin{aligned}
&\left\| \sum_{i=M+1}^{N} \hat{h}(\lambda_{i,N}) \langle \mathbf{P}_N f, \boldsymbol{\phi}_{i,N} \rangle \boldsymbol{\phi}_{i,N} \right\| \\
&\leq \sum_{i=M+1}^{N} (\lambda_{i,N}^{-d}) \left(\|f\|_{\mathcal{M}} + \left(\frac{\log(1/\delta)}{N} \right)^{\frac{1}{4}} \right) \quad (42) \\
&\leq \sum_{i=M+1}^{\infty} (\lambda_{i,N}^{-d}) \|f\|_{\mathcal{M}} \quad (43) \\
&\leq (1 + C_{\mathcal{M},1} \epsilon)^{-d} \sum_{i=M+1}^{\infty} (\lambda_i^{-d}) \|f\|_{\mathcal{M}} \quad (44) \\
&\leq M^{-1} \|f\|_{\mathcal{M}} := A_4(M). \quad (45)
\end{aligned}$$

We note that the bound is made up by terms $A_1(N) + A_2(M, N) + A_3(N) + A_4(M)$, related to the bandwidth of manifold signal M and the number of sampled points N. As ϵ scales with the order $\left(\frac{\log(CN/\delta)}{N} \right)^{\frac{1}{d+4}}$. This makes the bound scale with the order

$$\begin{aligned}
&\|\mathbf{h}(\mathbf{L}_N) \mathbf{P}_N f - \mathbf{P}_N \mathbf{h}(\mathcal{L}_\rho) f\| \\
&\leq C_1 \left(\frac{\log \frac{C_1 N}{\delta}}{N} \right)^{\frac{1}{d+4}} + C_2 \left(\frac{\log \frac{C_1 N}{\delta}}{N} \right)^{\frac{1}{d+4}} \theta_M^{-1} \\
&\qquad + C_3 \sqrt{\frac{\log(1/\delta)}{N}} + C_4 M^{-1}, \quad (46)
\end{aligned}$$

with $C_1 = C_{\mathcal{M},1} \frac{\pi^2}{6} \|f\|_{\mathcal{M}}$, $C_2 = C_{\mathcal{M},2} \frac{\pi^2}{6}$, $C_3 = \frac{\pi^2}{6}$ and $C_4 = \|f\|_{\mathcal{M}}$. As N goes to infinity, for every $\delta > 0$, there exists some M_0, such that for all $M > M_0$ it holds that $A_4(M) \leq \delta/2$. There also exists n_0, such that for all $N > n_0$, it holds that $A_1(N) + A_2(M_0, N) + A_3(N) \leq \delta/2$. We can conclude that the summations converge as N goes to infinity.

B. Proof of Corollary 2

We focus on deriving this expectation depending on the difference bound in probability in (16). We denote $\hat{\mathbf{y}}_N = \boldsymbol{\Phi}(\mathbf{H}, \mathbf{L}_N, \mathbf{P}_N f)$ and $\hat{g} = \boldsymbol{\Phi}(\mathbf{H}, \mathcal{L}_\rho, f)$. We have

$$\begin{aligned}
&\mathbb{P}\Bigg(\|\hat{\mathbf{y}}_N - \mathbf{P}_N \hat{g}\| \leq (C_1 + C_2 \theta_M) N^{-\frac{1}{d+4}} \left(\log \frac{1}{\delta} \right)^{\frac{1}{d+4}} + C_4 M^{-1} \\
&+ (C_1 + C_2 \theta_M) \left(\frac{\log N}{N} \right)^{\frac{1}{d+4}} + C_3 N^{-\frac{1}{2}} \sqrt{\log \frac{1}{\delta}} \Bigg) \geq 1 - 2\delta, \quad (47)
\end{aligned}$$

with $N \geq \log \frac{C}{\delta}$. We denote $k^2 = \log 1/\delta$, i.e. $\delta = e^{-k^2}$. Decomposing the expectation, we have

$$\begin{aligned}
&\mathbb{E}[\|\hat{\mathbf{y}}_N - \mathbf{P}_N \hat{g}\|_2] \leq \sum_{k=0}^{\sqrt{N/C_2}} \mathbb{P}\Bigg((C_1 + C_2 \theta_M)) N^{-\frac{1}{d+4}} k^{\frac{2}{d+4}} + \\
&C_3 N^{-\frac{1}{2}} k + (C_1 + C_2 \theta_M) \left(\frac{\log N}{N} \right)^{\frac{1}{d+4}} + C_4 M^{-1} \leq \|\hat{\mathbf{y}}_N - \mathbf{P}_N \hat{g}\| \\
&\leq (C_1 + C_2 \theta_M) N^{-\frac{1}{d+4}} (k+1)^{\frac{2}{d+4}} + C_3 N^{-\frac{1}{2}} (k+1) + C_4 M^{-1} \\
&+ (C_1 + C_2 \theta_M) \left(\frac{\log N}{N} \right)^{\frac{1}{d+4}} \Bigg) \Bigg((C_1 + C_2 \theta_M) N^{-\frac{1}{d+4}} (k+1)^{\frac{2}{d+4}} \\
&+ (C_1 + C_2 \theta_M) \left(\frac{\log N}{N} \right)^{\frac{1}{d+4}} + C_3 N^{-\frac{1}{2}} (k+1) + C_4 M^{-1} \Bigg) + \\
&\sum_{k=\sqrt{N/C_2}}^{\infty} \mathbb{P}\Bigg((C_1 + C_2 \theta_M)) N^{-\frac{1}{d+4}} k^{\frac{2}{d+4}} + C_3 N^{-\frac{1}{2}} k + C_4 M^{-1} \\
&+ (C_1 + C_2 \theta_M) \left(\frac{\log N}{N} \right)^{\frac{1}{d+4}} \leq \|\hat{\mathbf{y}}_N - \mathbf{P}_N \hat{g}\|_2 \leq C_4 M^{-1} + \\
&(C_1 + C_2 \theta_M) N^{-\frac{1}{d+4}} (k+1)^{\frac{2}{d+4}} + C_3 N^{-\frac{1}{2}} (k+1) + \\
&(C_1 + C_2 \theta_M) \left(\frac{\log N}{N} \right)^{\frac{1}{d+4}} \Bigg) \bar{M},
\end{aligned}$$

The upper bound of $\|\hat{\mathbf{y}}_N - \mathbf{P}_N\hat{g}\|_2$ can be derived with the norm of the output function of GNN when the GNN contains a single layer

$$\|\hat{\mathbf{y}}_N - \mathbf{P}_N\hat{g}\|_2 \leq \|\mathbf{\Phi}(\mathbf{H}, \mathbf{L}, \mathbf{P}_N f)\|_2 + \|\mathbf{P}_N\mathbf{\Phi}(\mathbf{H}, \mathcal{L}, f)\|_2 \tag{48}$$

$$\leq 2\|\mathbf{P}_N f\|_2. \tag{49}$$

Therefore the probability is zero when $(C_1+C_2\theta_M)N^{-\frac{1}{d+4}}(k+1)^{\frac{2}{d+4}} + C_3N^{-\frac{1}{2}}(k+1) > \bar{M} = 2\|\mathbf{P}_N f\|_2$, i.e. $k > \sqrt{N}\bar{M}$. Then we have

$$\begin{aligned}
&\mathbb{E}_\mu^N[\|\hat{\mathbf{y}}_N - \mathbf{P}_N\hat{g}\|_2] \\
&\leq \sum_{k=0}^{\sqrt{N/C}} 2e^{-k^2}\Bigg((C_1 + C_2\theta_M)N^{-\frac{1}{d+4}}(k+1)^{\frac{2}{d+4}} + \\
&(C_1 + C_2\theta_M)\left(\frac{\log N}{N}\right)^{\frac{1}{d+4}} + C_3N^{-\frac{1}{2}}(k+1) + C_4M^{-1}\Bigg) \\
&\qquad + \sum_{k=\sqrt{N/C}}^{\sqrt{N}\bar{M}} \bar{M}2e^{-N/C} \qquad (50) \\
&\leq C'N^{-\frac{1}{d+4}} + C''N^{-\frac{1}{2}} + C'''\left(\frac{\log N}{N}\right)^{\frac{1}{d+4}} + \bar{M}^2e^{-N/C}\sqrt{N} \qquad (51)
\end{aligned}$$

C. Proof of Theorem 1

Suppose $\mathbf{H}_E \in \arg\min_{\mathbf{H}\in\mathcal{H}} R_E(\mathbf{H})$, we have

$$\begin{aligned}
GA &\leq R_S(\mathbf{H}_E) - R_E(\mathbf{H}_E), \qquad (52)\\
&= \mathbb{E}_{X^N\sim\mu^N}\left[\ell\left(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N\right)\right] - \ell(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N) \qquad (53)
\end{aligned}$$

We can now add and subtract $\ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g)$, as follows

$$\begin{aligned}
GA \leq &\left(\mathbb{E}_{X^N\sim\mu^N}\left[\ell\left(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N\right)\right] - \ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g)\right) \\
&+ \left(\ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g) - \ell(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N)\right) \qquad (54)
\end{aligned}$$

As we consider the L^2 loss function, therefore, by Cauchy-Schwartz we have,

$$\begin{aligned}
GA \leq &\underbrace{\left|\mathbb{E}_{X^N\sim\mu^N}\left[\ell\left(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N\right)\right] - \ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g)\right|}_{\textbf{LHS}} \\
&+ \underbrace{\left|\ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g) - \ell(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N)\right|}_{\textbf{RHS}}. \qquad (55)
\end{aligned}$$

We assume the loss function to be L_2 loss. The **LHS** in (55) can be decomposed as

$$\begin{aligned}
&\left|\ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g) - \mathbb{E}_{X_N\sim\mu^N}\left[\ell\left(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N\right)\right]\right| \\
&= \left|\|\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - g\|_{\mathcal{M}} - \mathbb{E}_{\mu^N}[\|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{y}_N\|_2]\right| \qquad (56)\\
&\leq \left|\mathbb{E}_{\mu^N}[\|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f)\|_2] + \right. \qquad (57)\\
&\left.\mathbb{E}_\mu[\|\mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - \mathbf{P}_N g\|_2] - \|\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - g\|_{\mathcal{M}}\right| \\
&= \mathbb{E}_{\mu^N}[\|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f)\|_2], \qquad (58)
\end{aligned}$$

which is the conclusion in Corollary 2.

For the **RHS** in (55), we have

$$\begin{aligned}
&\left|\ell(\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N), \mathbf{y}_N) - \ell(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f), g)\right| \\
&\leq \left|\|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{P}_N g\|_2 - \|\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - g\|_{\mathcal{M}}\right| \qquad (59)\\
&\leq \left|\|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f)\|_2 + \right. \\
&\qquad \left.\|\mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - \mathbf{P}_N g\|_2 - \|\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - g\|_{\mathcal{M}}\right| \qquad (60)\\
&\leq \|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f)\|_2 + \\
&\left|\|\mathbf{P}_N(\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - g)\|_2 - \|\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f) - g\|_{\mathcal{M}}\right| \qquad (61)\\
&\leq \|\mathbf{\Phi}(\mathbf{H}_E, \mathbf{x}_N) - \mathbf{P}_N\mathbf{\Phi}(\mathbf{H}_E, \mathcal{L}, f)\|_2 + O\left(\sqrt{\frac{\log(1/\delta)}{N}}\right). \qquad (62)
\end{aligned}$$

The first part in (62) is bounded by Proposition 1. Combining these two parts, we can achieve the conclusion that

$$GA = \mathcal{O}\left(\left(\frac{\log\frac{N}{\delta}}{N}\right)^{\frac{1}{d+4}} + \left(\frac{\log N}{N}\right)^{\frac{1}{d+4}}\right). \tag{63}$$