Uncertainty-Aware Message Passing Neural Networks

Alesia Chernikova[1,2], **Moritz Laber**[2], **Narayan G. Sabhahit**[2], **Tina Eliassi-Rad**[1,2]
[1]Khoury College of Computer Sciences [2]Network Science Institute
Northeastern University, Boston, MA, USA
{a.chernikova, laber.m, sabhahit.n, eliassi}@northeastern.edu

Abstract

Existing theoretical guarantees for message passing neural networks (MPNNs) assume deterministic node features, whereas in this work we address the more realistic setting where inherent noise or finite measurement precision leads to uncertainty in node features. We assume node features are multivariate Gaussian distributions and propagate their first and second moments through the MPNN architecture. We employ Polynomial Chaos Expansion to approximate nonlinearities, and use the resulting node embedding distributions to analytically produce probabilistic node-wise robustness certificates against L_2-bounded node feature perturbations. Moreover, we model node features as multivariate random variables and introduce *Feature Convolution Distance*, FCD_p, a Wasserstein distance-based pseudometric that matches the discriminative power of node-level MPNNs. We show that MPNNs are globally Lipschitz continuous functions with respect to FCD_p. Our framework subsumes the deterministic case via Dirac measures and provides a foundation for reasoning about algorithmic stability in MPNNs with uncertainty in node features.

1 Introduction

Message-passing Neural Networks are the predominant method for applying machine learning to graph-structured data due to their strong performance on graph-, edge-, and node-level tasks [Chami et al., 2022, Hu et al., 2020, Hamilton, 2020]. MPNNs take node feature vectors as input and use a recursive neighborhood aggregation scheme to produce node embeddings that capture structural information from the graph [Gilmer et al., 2017, Scarselli et al., 2008].

Most existing MPNN formulations assume that node features are deterministic vectors. In practice, however, features are often uncertain due to inherent noise, finite measurement precision, and/or coarse graining, making a framework for uncertainty quantification in MPNNs necessary [Wang et al., 2024, Zhang et al., 2024]. We assume that node features are multivariate Gaussian distributions and use moment propagation to track uncertainty throughout the architecture. Gaussian distributions are fully characterized by their first two moments: the mean vector and the covariance matrix. We propagate these moments exactly through linear message-passing operations and approximately through nonlinearities using Polynomial Chaos Expansion (PCE) [Wiener, 1938, Ghanem and Spanos, 1991, Sullivan, 2015, Soize, 2017], a representation of functions of random variables as an expansion in orthogonal polynomials of other random variables. Thereby, we obtain node embedding distributions. Using the node embedding distributions, we establish probabilistic node-wise robustness certificates for L_2-bounded node feature perturbations.

Moreover, we define *Feature Convolution Distance*, FCD_p, in the input space of nodes to analyze the theoretical performance of node-level MPNNs with stochastic node features. FCD_p is a Wasserstein distance-based pseudometric. That is, it satisfies the non-negativity, symmetry, and triangle inequality axioms, but not the identity of indiscernibles. Focusing on the Simple Graph Convolution (SGC) model [Wu et al., 2019]—an MPNN without nonlinearities that balances efficiency, interpretability,

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: .

and performance—we show that FCD_p achieves discriminative power equivalent to SGC with and without output nonlinearity. By establishing Lipschitz continuity for the SGC architecture with respect to FCD_p, we lay the foundation for a rigorous analysis of generalization guarantees based on the algorithmic robustness property. Notably, our framework remains consistent when handling deterministic features, thereby unifying stochastic and deterministic scenarios under a single theoretical framework.

Our **contributions** are as follows.

- We introduce uncertainty-aware message passing neural networks. These MPNNs handle uncertainty in node features, where the features are multivariate Gaussian distributions.
- We represent node-feature distributions by their first and second moments and propagate these moments through the MPNN. We approximate nonlinearities using Polynomial Chaos Expansion (PCE). We obtain probabilistic robustness guarantees at the node level by deriving per-node certified radii against L_2-bounded feature perturbation attacks.
- We introduce *Feature Convolution Distance*, FCD_p, a Wasserstein distance-based pseudometric for comparing nodes in the MPNN input space. We argue that, to effectively support theoretical analysis, such a distance must incorporate the structural updates produced by the message-passing process. We demonstrate how the proposed distance applies when there is no uncertainty about node features (i.e., when they are deterministic vectors).
- We demonstrate that FCD_p provides global Lipschitz continuity between the input space and the embedding space, and exhibits the same discriminative power as the p-Wasserstein distance between probability distributions of random variables transformed by Simple Graph Convolution (SGC) with and without nonlinearity.

2 Background

2.1 Graph Convolutional Networks and Their Variants

The Graph Convolutional Network (GCN) [Kipf and Welling, 2017] is a widely used and effective MPNN architecture. A GCN with L layers on a graph with n nodes is defined by the following recursion:

$$X^{(l+1)} = \sigma(SX^{(l)}W^{(l)}), \tag{1}$$

where $S = (I_n + D)^{-1/2}(I_n + A)(I_n + D)^{-1/2} \in \mathbb{R}^{n\times n}$ is the structural update matrix defined in terms of the adjacency matrix $A \in \mathbb{R}^{n\times n}$, the diagonal matrix of node degrees $D = \mathrm{diag}\{d_1, \cdots, d_n\} \in \mathbb{R}^{n\times n}$, and the n dimensional identity matrix I_n. We denote the lth layer's weight matrix as $W^{(l)} \in \mathbb{R}^{f_l \times f_{l+1}}$. The matrix $X^{(l)} \in \mathbb{R}^{n\times f_l}$ gathers the f_l-dimensional embeddings at layer l or the input features at $l = 0$. The pointwise nonlinearity σ can be any Lipschitz continuous function (such as ReLU).

In the absence of nonlinearities, all weight matrices can be combined into a single matrix $W \in \mathbb{R}^{f_0\times f_L}$. The resulting architecture is called the Simplified Graph Convolution (SGC), [Wu et al., 2019]:

$$X^{(L)} = \Theta(X^{(0)}) = S^L X^{(0)} W. \tag{2}$$

In this work, we also consider a variant of SGC with a single output nonlinearity $\sigma(\cdot)$. That is, $\hat{\Theta}(\cdot) = \sigma(\Theta(\cdot))$. This variant is useful when SGC is applied to node classification tasks.

2.2 Polynomial Chaos Expansion (PCE)

The Polynomial Chaos Expansion (PCE) [Wiener, 1938, Ghanem and Spanos, 1991] is a well established technique for uncertainty quantification in physics and engineering [Sullivan, 2015, Soize, 2017].

Let $\mathbf{f} : \mathbb{R}^f \to \mathbb{R}^{f'}$ be a vector-valued function of the random variable $\boldsymbol{\xi} \in \mathbb{R}^f$, which is a multivariate Gaussian random variable with mean $\mu_{\boldsymbol{\xi}}$ and covariance $\Sigma_{\boldsymbol{\xi}}$. PCE represents $\mathbf{f}(\boldsymbol{\xi})$ as an expansion in

Hermite polynomials $\{\Phi_k\}_{k=0}^{\infty}$. That is,

$$\mathbf{f}(\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \boldsymbol{c}_k \, \Phi_k(\boldsymbol{\xi}) \approx \sum_{k=0}^{K} \boldsymbol{c}_k \, \Phi_k(\boldsymbol{\xi}), \tag{3}$$

where $\boldsymbol{c}_k \in \mathbb{R}^{f'}$ are coefficient vectors obtained by Galerkin projection $\boldsymbol{c}_k = \langle \mathbf{f}, \Phi_k \rangle$ with respect to the Gaussian density $p_{\boldsymbol{\xi}}$.

While extensions beyond the Gaussian case like generalized Polynomial Chaos (gPC) [Askey and Wilson, 1985, Xiu and Karniadakis, 2002, 2003] and arbitrary Polynomial Chaos (aPC) [Oladyshkin and Nowak, 2012, Navarro et al., 2014, Paulson et al., 2017] exist, they do not have additional benefits if correlations are linear and do not scale to the high-dimensional inputs commonly encountered in machine learning when correlations are nonlinear.

2.3 Wasserstein Distance

The field of optimal transport [Villani et al., 2008] defines the p-Wasserstein distance, W_p, for $p \in [1, \infty)$ between two distributions $\mathbb{P}_{\boldsymbol{\xi}}$ on $\mathbb{R}^f$ and $\mathbb{P}_{\boldsymbol{\xi'}}$ on $\mathbb{R}^{f'}$ as follows:

$$W_p(\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}}) = \inf_{\gamma \in \Gamma(\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}})} \left(\iint_{\mathbb{R}^f \times \mathbb{R}^{f'}} \|\boldsymbol{\xi} - \boldsymbol{\xi'}\|^p \, d\gamma(\boldsymbol{\xi}, \boldsymbol{\xi'}) \right)^{1/p}, \tag{4}$$

where γ is a coupling, i.e., a distribution on $\mathbb{R}^{f \times f'}$ that has $\mathbb{P}_{\boldsymbol{\xi}}$ and $\mathbb{P}_{\boldsymbol{\xi'}}$ as its marginals. In the deterministic case when $\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}}$ are Dirac measures, the Wasserstein distance reduces to the L_p-norm.

3 Uncertainty Quantification by Propagating Moments of Distributions

Our node features are multivariate Gaussian distributions. We provide expressions for the propagation of the first and second moments of these distributions through a GCN. We consider node embeddings at each layer l (including input features $l = 0$) as random variables $\boldsymbol{\xi}_i^{(l)} \in \mathbb{R}^{f_l}$ for each node $v_i \in V$. For ease of notation, we define the vector $\vec{\boldsymbol{\xi}}^{(l)} = [\boldsymbol{\xi}_1^{(l)}, \ldots, \boldsymbol{\xi}_n^{(l)}]^\mathsf{T} \in \mathbb{R}^{nf_l}$ by concatenation. This setup allows for linear correlation between features at the same node, as well as between features at different nodes.

The features $\vec{\boldsymbol{\xi}}^{(l),s}$ after the **structural update** are

$$\vec{\boldsymbol{\xi}}^{(l),s} = (S \otimes I_{f_{l-1}}) \vec{\boldsymbol{\xi}}^{(l-1)}, \tag{5}$$

where $I_{f_{l-1}}$ is the f_{l-1}-dimensional identity matrix and S was defined in Section 2.1. The random variable $\vec{\boldsymbol{\xi}}^{(l),s}$ has the following mean $\mu_{\vec{\boldsymbol{\xi}}^{(l),s}}$ and covariance $\Sigma_{\vec{\boldsymbol{\xi}}^{(l),s}}$:

$$\mu_{\vec{\boldsymbol{\xi}}^{(l),s}} = (S \otimes I_{f_{l-1}}) \mu_{\vec{\boldsymbol{\xi}}^{(l-1)}} \tag{6}$$

$$\Sigma_{\vec{\boldsymbol{\xi}}^{(l),s}} = (S \otimes I_{f_{l-1}}) \Sigma_{\vec{\boldsymbol{\xi}}^{(l-1)}} (S \otimes I_{f_{l-1}})^\mathsf{T} \tag{7}$$

where $\mu_{\vec{\boldsymbol{\xi}}^{(l-1)}}$ and $\Sigma_{\vec{\boldsymbol{\xi}}^{(l-1)}}$ are the mean and covariance of the features $\vec{\boldsymbol{\xi}}^{(l-1)}$ at the previous layer.

The **weight update** yields the features $\vec{\boldsymbol{\xi}}^{(l),w}$ defined by

$$\vec{\boldsymbol{\xi}}^{(l),w} = (I_n \otimes W^{(l)}) \vec{\boldsymbol{\xi}}^{(l),s}, \tag{8}$$

where I_n is the n-dimensional identity matrix. This random variable has the following mean and covariance:

$$\mu_{\vec{\boldsymbol{\xi}}^{(l),w}} = (I_n \otimes W^{(l)}) \mu_{\vec{\boldsymbol{\xi}}^{(l),s}} \tag{9}$$

$$\Sigma_{\vec{\boldsymbol{\xi}}^{(l),w}} = (I_n \otimes W^{(l)}) \Sigma_{\vec{\boldsymbol{\xi}}^{(l),s}} (I_n \otimes W^{(l)})^\mathsf{T}. \tag{10}$$

The features $\vec{\boldsymbol{\xi}}^{(l),e}$ after the **nonlinear update** are as follows:

$$\vec{\boldsymbol{\xi}}^{(l),e} = \sigma\left(\vec{\boldsymbol{\xi}}^{(l),w}\right), \tag{11}$$

where the nonlinear function σ is applied elementwise. The mean $\mu_{\vec{\boldsymbol{\xi}}^{(l),e}}$ and covariance $\Sigma_{\vec{\boldsymbol{\xi}}^{(l),e}}$ can be obtained by PCE as follows:

$$\mu_{\vec{\boldsymbol{\xi}}^{(l),e}} = \boldsymbol{c}_0^{(l)} \tag{12}$$

$$\Sigma_{\vec{\boldsymbol{\xi}}^{(l),e}} = \sum_{k=1}^{K} \boldsymbol{c}_k^{(l)} \boldsymbol{c}_k^{(l)\mathsf{T}} \tag{13}$$

$c_k^{(l)} \in \mathbb{R}^{nf_l}$ are the PCE coefficients. We refer the reader to Appendix B for their derivation.

Note that after applying the first-layer nonlinearity, the derived mean and covariance are no longer from a Gaussian distribution. We approximate the distributions at layers $l > 1$ by performing Gaussian moment matching and assume that the resulting mean and covariance are from a Gaussian distribution. This approach is valid because the Gaussian distribution is the maximum entropy distribution given the first two moments.

4 Probabilistic Robustness Certification

Robustness to feature perturbations is important for graph machine learning architectures. Since we know the moments of distributions for the random variables representing the node embeddings at the last layer, we can certify node-wise robustness of MPNNs for node-classification task against L_2 feature perturbations.

Let $\boldsymbol{\xi}_i^z \in \mathbb{R}^{f_L}$ denote the random variable of node v_i's logit with mean $\mu_{\boldsymbol{\xi}_i^z}$ and covariance $\Sigma_{\boldsymbol{\xi}_i^z}$.

Theorem 1. *An MPNN for a node classification task is robust against feature perturbation $||\Delta||_2 = \epsilon$ with probability at least $1 - \delta$ if:*

$$\epsilon < \min_{y \neq y^\star} \frac{\hat{\mu}_{\boldsymbol{\xi}_{iy}^z} - \sqrt{\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z}} \sqrt{\frac{1-\delta_y}{\delta_y}}}{\sqrt{2}C}, \tag{14}$$

where C is the Lipschitz constant of the GCN w.r.t. individual L_2 node feature perturbation. δ_y is the probability of misclassifying node v_i to class y and $\delta = \sum_y \delta_y$. The mean and covariance of the random margin between the logit element associated with true label class $y^\star$ and logit element associated with any other class label $y \neq y^\star$ are

$$\hat{\mu}_{\boldsymbol{\xi}_{iy}^z} = \mu_{\boldsymbol{\xi}_{i,y^\star}^z} - \mu_{\boldsymbol{\xi}_{i,y}^z} \tag{15}$$

$$\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z} = \Sigma_{\boldsymbol{\xi}_i^z, y^\star y^\star} + \Sigma_{\boldsymbol{\xi}_i^z, yy} - 2\,\Sigma_{\boldsymbol{\xi}_i^z, y^\star y} \tag{16}$$

We refer the reader to Appendix C for the proof of Theorem 1.

5 Wasserstein-based Pseudometric

We introduce a new distance FCD_p between nodes $v_i, v_j \in V$ of the graph $G = (V, E)$ with associated random variables $\boldsymbol{\xi}_i^{(0)}, \boldsymbol{\xi}_j^{(0)}$ for the SGC architecture as follows:

$$\mathrm{FCD}_p\,(v_i, v_j) := W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = \inf_{\gamma \in \Gamma(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s})} \left(\iint_{\mathbb{R}^{f_0} \times \mathbb{R}^{f_0}} \|\boldsymbol{\xi}_i^s - \boldsymbol{\xi}_j^s\|^p \, d\gamma(\boldsymbol{\xi}_i^s, \boldsymbol{\xi}_j^s) \right)^{1/p} \tag{17}$$

where $\boldsymbol{\xi}_i^s = [S^L \boldsymbol{\xi}^{(0)}]_i$ is the random variable associated with node v_i after the structural update step of SGC.

Corollary 1. *When the input node features follow multivariate Gaussian distributions* $\boldsymbol{\xi}_i^{(0)} \sim \mathcal{N}(\mu_i, \Sigma_i)$ *for all* $v_i \in V$*, the node-wise random variables after structural update are also Gaussian distributions* $\boldsymbol{\xi}_i^s \sim \mathcal{N}(\mu_i^s, \Sigma_i^s)$*. For* $p = 2$*, the* FCD_p *has the following analytical form:*

$$\mathrm{FCD}_p(v_i, v_j) \ = W_2(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = \sqrt{\|\mu_i^s - \mu_j^s\|_2^2 + \mathrm{Tr}\left(\Sigma_i^s + \Sigma_j^s - 2\left(\left(\Sigma_i^s\right)^{1/2}\Sigma_j^s\left(\Sigma_i^s\right)^{1/2}\right)^{1/2}\right)}, \tag{18}$$

where $\mathrm{Tr}[\cdot]$ *is the trace of a matrix.*

For the other values of p or other probability distributions, numerical methods such as the Sinkhorn algorithm [Sinkhorn and Knopp, 1967, Cuturi, 2013] can be used. In the deterministic scenario (i.e., when the probability measures are Dirac measures), the distance is equal to the following:

$$\mathrm{FCD}_p(v_i, v_j) = \left|\left|[S^L X^{(0)}]_i - [S^L X^{(0)}]_j|\right|\right|_p. \tag{19}$$

5.1 Characteristics of FCD_p

Proposition 1. *The distance from a node* v_i *to itself is zero:* $\mathrm{FCD}_p(v_i, v_i) = 0$*.*

This follows from the fact that the probability distributions associated with the same random variable $\boldsymbol{\xi}_i$ are identical.

Proposition 2. *The distance from* v_i *to* v_j *is always the same as the distance from* v_j *to* v_i*:* $\mathrm{FCD}_p(v_i, v_j) = \mathrm{FCD}_p(v_j, v_i)$*.*

FCD_p is a Wasserstein-based distance. Thus, this property automatically follows from the property of the Wasserstein distance.

Proposition 3. $\mathrm{FCD}_p(v_i, v_j) \leq \mathrm{FCD}_p(v_i, v_k) + \mathrm{FCD}_p(v_k, v_j)$*.*

This property is also a direct consequence of the properties of the Wasserstein distance.

FCD_p is a pseudometric because it does not satisfy the identity of indiscernibles. That is, FCD_p does not guarantee that it is always positive between two distinct points. $\mathrm{FCD}_p(v_i, v_j)$ may be equal to zero when v_i is different from v_j.

5.2 Discriminative Power

If a distance has the same discriminative power as an architecture, data points that have non-zero distance in the input space will have non-zero distance in the output space. Next, we show that FCD_p has the same discriminative power as SGC.

Theorem 2. FCD_p *has the same discriminative power as* $\hat{\Theta}$*:*

$$W_p(\hat{\Theta}(v_i), \hat{\Theta}(v_j)) > 0 \Rightarrow \mathrm{FCD}_p(v_i, v_j) > 0 \tag{20}$$

We refer the reader to Appendix D for the proof of Theorem 2.

Corollary 2. *In the case of Dirac measures, the discriminative power of* FCD_p *remains valid and can be expressed as follows:*

$$\left|\left|[\sigma(S^L X^{(0)} W)]_i - [\sigma(S^L X^{(0)} W)]_j\right|\right|_p > 0 \Rightarrow \left|\left|[S^L X^{(0)}]_i - [S^L X^{(0)}]_j\right|\right|_p > 0 \tag{21}$$

6 Lipschitz Continuity

Lipschitz continuity of SGC allows us to reason how distances in the input space translate into distances in the embedding space. Moreover, it provides the basis for reasoning about the algorithmic robustness of the SGC; and consequently, reasoning about the generalization abilities of the model [Xu and Mannor, 2012].

Theorem 3. *SGC $\hat{\Theta}$ is a globally Lipschitz function w.r.t* FCD_p*:*

$$W_p(\hat{\Theta}(v_i), \hat{\Theta}(v_j)) \leq C_L \cdot \mathrm{FCD}_p(v_i, v_j) \tag{22}$$

We refer the reader to Appendix E for the proof of Theorem 3. Theoretical guarantees for the default Θ architecture (i.e., SGC without output nonlinearity) can be found in Appendices D and E.

Corollary 3. *In the case of the Dirac measures, the Lipschitz continuity of $\hat{\Theta}$ is valid and can be expressed as follows:*

$$\left|\left|[\sigma(S^L XW)]_i - [\sigma(S^L XW)]_j\right|\right|_p \leq C_L \left|\left|[S^L X]_i - [S^L X]_j\right|\right|_p \tag{23}$$

Experimental Results. We present results examining the Lipschitz continuity of SGC models in transductive node-classification tasks. We train 2-layer and 3-layer SGC models with ReLU nonlinearities on the standard benchmark datasets of Cora, Citeseer, and Pubmed [Yang et al., 2016]. The training phase uses the Adam optimizer [Kingma and Ba, 2015] with a learning rate of 0.001 and a weight decay parameter of 0.0005, over 200 epochs.

We evaluate Lipschitz continuity by computing pairwise graph distances between nodes drawn from the test sets of each dataset, along with the corresponding 2-Wasserstein distances between the respective node embeddings. To model uncertainty in node features, we generate 10 realizations for each node by adding Gaussian noise $\epsilon \sim \mathcal{N}(0, \Sigma)$. We measure the correlation between distances, following [Vasileiou et al., 2025].

Figures 1 and 2, respectively, show the scatter plots of FCD_p on nodes versus the W_2 distance on node embeddings for a 2-layer and a 3-layer SGC model on the Cora, Citeseer, and Pubmed datasets [Yang et al., 2016] in the transductive node-classification task. While there is no linear dependency, our experiments do not contradict the Lipschitzness of SGC. The Lipschitz constant can be upper-bounded by the slope of the line through (0, 0) that lies above all observed data points [Vasileiou et al., 2025].

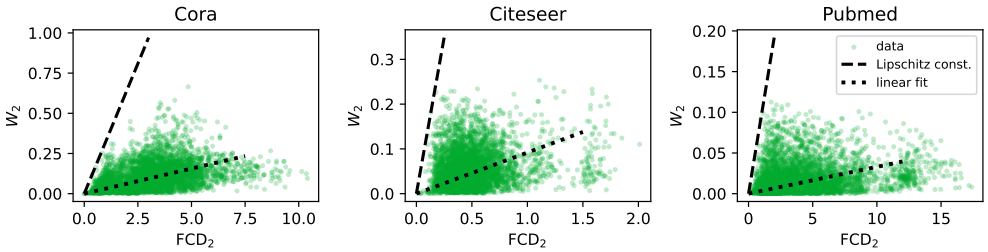

Figure 1: Scatter plots between FCD_p with $p = 2$ on nodes and W_2 distance on node embeddings of 2-layer SGC models on Cora (Pearson correlation = 0.43, Lipschitz constant = 0.32), Citeseer (Pearson correlation = 0.2, Lipschitz constant = 1.39), and Pubmed (Pearson correlation = 0.19, Lipschitz constant = 0.1) datasets. The Pearson correlation shows that FCD_p in the input space is positively correlated to the W_2 distance in the embedding space.

7 Related Work

Uncertainty Quantification in MPNNs. Uncertainty quantification has recently seen a surge of attention in graph machine learning [Wang et al., 2024] and deep learning [Hüllermeier and Waegeman, 2021, Gawlikowski et al., 2023] leading to a wide variety of techniques aimed at quantifying both epistemic and aleatoric uncertainty. The problem of tracing uncertainty through nonlinear models has a long tradition in science and engineering [Sullivan, 2015, Soize, 2017]. Polynomial chaos expansion (PCE) is one particularly well-developed technique originally introduced for Gaussian random variables [Wiener, 1938, Ghanem and Spanos, 1991] and later extended to arbitrary distributions as generalized polynomial chaos (gPC) [Xiu and Karniadakis, 2002, 2003] and to distributions that are only accessible in terms of their moments as arbitrary polynomial chaos (aPC) [Oladyshkin and Nowak, 2012, Navarro et al., 2014, Paulson et al., 2017]. Only recently have these techniques found applications in machine learning [Du, 2025]. To the best of our knowledge, we are the first to apply PCE for uncertainty quantification in MPNNs. We refer the reader to Appendix B for details.

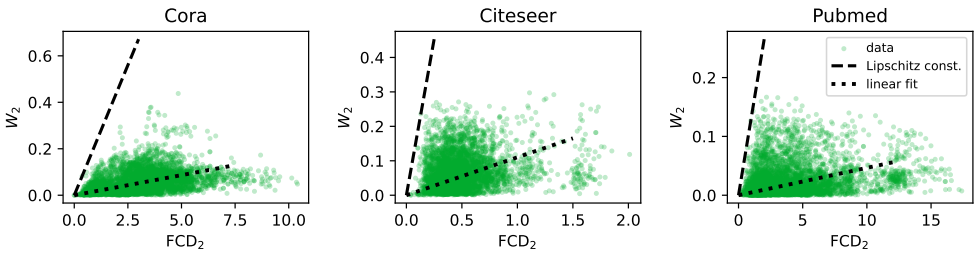

Figure 2: Scatter plots between FCD_p with $p = 2$ on nodes and W_2 distance on node embeddings of 3-layer SGC models on Cora (Pearson correlation = 0.39, Lipschitz constant = 0.33), Citeseer (Pearson correlation = 0.20, Lipschitz constant = 1.81), and Pubmed (Pearson correlation = 0.18, Lipschitz constant = 0.13) datasets. Similar to Figure 1, the Pearson correlation shows that FCD_p in the input space is positively correlated to the W_2 distance in the embedding space.

Robustness Certification. Cohen et al. [2019] establish that a classifier smoothed with Gaussian noise yields provable L_2 radius guarantees. Certification is conducted via sampling-based estimation. In a follow-up work, Kumar et al. [2020] propose a method to generate certified radii for the prediction confidence of a smoothed classifier. Pautov et al. [2022] introduce the CC-Cert framework, which leverages concentration inequalities to certify the robustness of neural networks under input perturbations. Zügner and Günnemann [2019] introduce one of the first certification schemes for GCNs to defend against node-feature modifications under L_0-bounded budgets. GNNCert [Yang et al., 2024] provides deterministic certification for graph classification against both structure and feature perturbations by guaranteeing label invariance when the numbers of modified edges and node features are bounded. Our work diverges from existing approaches by modeling uncertainty in node features directly by propagating the moments of node feature distributions through the architecture and converts logit-level moment information into probabilistic node-wise robustness certificates. This complements deterministic graph certificates and sampling-based smoothing guarantees.

Pseudometrics. Rauchwerger et al. [2024] extend iterated degree measures to graphon-signals, and show compactness of the resulting space, establishing Lipschitz continuity and universal approximation for MPNNs. In [Levie, 2024], a one-sided Lipschitz inequality is proven, bounding feature distances by the graphon-signal cut distance, though without universal approximation and with slow generalization rates. In [Chuang and Jegelka, 2022], the Tree Mover's Distance (TMD) is proposed for graphs with features, relating it to generalization under distribution shifts, but lacking universal approximation. In [Chen et al., 2022, 2023], MPNNs are shown to separate points and be Lipschitz over WL distances, with universal approximation only on compact subspaces since the full space is not compact. Vasileiou et al. [2024] leverage graph similarity theory to assess the influence of graph structure, aggregation, and loss functions on MPNN generalization abilities. Vasileiou et al. [2025] is the closest to our work, where the authors introduce a unified framework for analyzing the generalization properties of MPNNs in inductive and transductive node and link prediction tasks while relaxing nodes i.i.d. assumptions; but they do not consider the case when there is uncertainty in node features.

8 Conclusion

We introduced a framework for uncertainty propagation in MPNNs, showing exact moment propagation through linear layers and tractable approximations for nonlinearities via PCE. In addition, we proposed a Wasserstein distance-based pseudometric FCD_p that unifies the characteristics of deterministic and stochastic node features, matches the discriminative power of the SGC with and without nonlinearity, and allows us to derive Lipschitz continuity.

Future Work. First, we plan to extend our work on uncertainty quantification to node feature distributions beyond Gaussian. Second, existing theory on generalization bounds does not cover the case where the MPNN operates on random variables. FCD_p will be useful once we extend the theory on generalization bounds. A natural next step is to extend the covering number-based arguments [Xu and Mannor, 2012, Vasileiou et al., 2025] to our stochastic feature setting.

References

Richard Askey and James Arthur Wilson. *Some Basic Hypergeometric Orthogonal Polynomials that Generalize Jacobi Polynomials*. American Mathematical Society, 1985.

Ines Chami, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. Machine learning on graphs: A model and comprehensive taxonomy. *Journal of Machine Learning Research*, 23 (89):1–64, 2022.

Samantha Chen, Sunhyuk Lim, Facundo Memoli, Zhengchao Wan, and Yusu Wang. Weisfeiler-Lehman meets Gromov-Wasserstein. In *ICML*, pages 3371–3416, 2022.

Samantha Chen, Sunhyuk Lim, Facundo Memoli, Zhengchao Wan, and Yusu Wang. The Weisfeiler-Lehman distance: Reinterpretation and connection with gnns. In *Proceedings of 2nd Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML)*, pages 404–425, 2023.

Ching-Yao Chuang and Stefanie Jegelka. Tree mover's distance: Bridging graph metrics and stability of graph neural networks. *Advances in Neural Information Processing Systems*, 35:2944–2957, 2022.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *ICML*, pages 1310–1320, 2019.

Marco Cuturi. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Xiaoping Du. Uncertainty Quantification for Machine Learning-Based Prediction: A Polynomial Chaos Expansion Approach for Joint Model and Input Uncertainty Propagation. `http://arxiv.org/abs/2507.14782`, July 2025.

Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, Muhammad Shahzad, Wen Yang, Richard Bamler, and Xiao Xiang Zhu. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(1):1513–1589, 2023.

Roger G. Ghanem and Pol D. Spanos. Stochastic Finite Element Method: Response Statistics. In Roger G. Ghanem and Pol D. Spanos, editors, *Stochastic Finite Elements: A Spectral Approach*, pages 101–119. Springer, 1991.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *ICML*, pages 1263–1272, 2017.

William L. Hamilton. *Graph Representation Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool, 2020.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in Neural Information Processing Systems*, 33:22118–22133, 2020.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3):457–506, 2021.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

Aounon Kumar, Alexander Levine, Soheil Feizi, and Tom Goldstein. Certifying confidence via randomized smoothing. *Advances in Neural Information Processing Systems*, 33:5165–5177, 2020.

Ron Levie. A graphon-signal analysis of graph neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

Maria Navarro, Jeroen Witteveen, and Joke Blom. Polynomial Chaos Expansion for general multivariate distributions with correlated variables. http://arxiv.org/abs/1406.5483, June 2014.

S. Oladyshkin and W. Nowak. Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. *Reliability Engineering & System Safety*, 106:179–190, 2012.

Joel A Paulson, Edward A Buehler, and Ali Mesbah. Arbitrary polynomial chaos for uncertainty propagation of correlated random variables in dynamic systems. *IFAC-PapersOnLine*, 50(1): 3548–3553, 2017.

Mikhail Pautov, Nurislam Tursynbek, Marina Munkhoeva, Nikita Muravev, Aleksandr Petiushko, and Ivan Oseledets. Cc-cert: A probabilistic approach to certify general robustness of neural networks. In *AAAI*, pages 7975–7983, 2022.

Levi Rauchwerger, Stefanie Jegelka, and Ron Levie. Generalization, expressivity, and universality of graph neural networks on attributed graphs, 2024. URL https://arxiv.org/abs/2411.05464.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

Philip Schnabel. Polynomial chaos expansion for dependent inputs. Technical Report RSUQ-2023-001, ETH Zürich, January 2023. https://ethz.ch/content/dam/ethz/special-interest/baug/ibk/risk-safety-and-uncertainty-dam/publications/reports/RSUQ-2023-001.pdf.

Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

Christian Soize. *Uncertainty Quantification: An Accelerated Course with Advanced Applications in Computational Engineering*. Springer, 2017.

T.J. Sullivan. *Introduction to Uncertainty Quantification*. Springer, 2015.

Antonis Vasileiou, Ben Finkelshtein, Floris Geerts, Ron Levie, and Christopher Morris. Covered forest: Fine-grained generalization analysis of graph neural networks, 2024. URL https://arxiv.org/abs/2412.07106.

Antonis Vasileiou, Timo Stoll, and Christopher Morris. Understanding generalization in node and link prediction, 2025. URL https://arxiv.org/abs/2507.00927.

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2008.

Fangxin Wang, Yuqing Liu, Kay Liu, Yibo Wang, Sourav Medya, and Philip S. Yu. Uncertainty in Graph Neural Networks: A Survey. *Transactions on Machine Learning Research*, 2024.

Norbert Wiener. The Homogeneous Chaos. *American Journal of Mathematics*, 60(4):897–936, 1938.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *ICML*, pages 6861–6871, 2019.

Dongbin Xiu and George Em Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002.

Dongbin Xiu and George Em Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187(1):137–167, 2003.

Huan Xu and Shie Mannor. Robustness and generalization. *Machine learning*, 86:391–423, 2012.

Han Yang, Binghui Wang, Jinyuan Jia, et al. GNNCert: Deterministic certification of graph neural networks against adversarial perturbations. In *ICLR*, 2024.

Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *ICML*, pages 40–48, 2016.

He Zhang, Bang Wu, Xingliang Yuan, Shirui Pan, Hanghang Tong, and Jian Pei. Trustworthy graph neural networks: Aspects, methods, and trends. *Proceedings of the IEEE*, 112(2):97–139, February 2024.

Daniel Zügner and Stephan Günnemann. Certifiable robustness and robust training for graph convolutional networks. In *KDD*, pages 246–256, 2019.

A Notation and Additional Background Information

Table 1 lists the notation used throughout the paper. The rest of the section introduces graphs, functions of random variables, and coupling of random variables.

Symbol	Explanation
$G = (V, E)$	graph with nodes set V and edge set E.
$n = \|V\|$	number of nodes in the graph
$m = \|E\|$	number of edges in the graph
$v_i \in V$	node of the graph G
d_i	degree of node v_i
$I_n \in \mathbb{R}^{n \times n}$	n-dimensional identity matrix
$A \in \mathbb{R}^{n \times n}$	adjacency matrix of G
$D = \mathrm{diag}(d_1, \ldots, d_n)$	diagonal matrix of node degrees
$S = (I + D)^{-1/2}(I + A)(I + D)^{-1/2}$	structural update matrix
$s_{ij} = [S]_{ij}$	entries of S
$l \in \{0, \ldots, L\}$	MPNN layer index
f_l	node feature dimension in layer l
$\boldsymbol{x}_i^{(l)} \in \mathbb{R}^{1 \times f}$	deterministic node features associated with node v_i in layer l
$X^{(l)} \in \mathbb{R}^{n \times f}$	matrix of deterministic node features in layer l
$W^{(l)} \in \mathbb{R}^{f_l \times f_{l+1}}$	weight matrix of layer l.
$\sigma : \mathbb{R} \to \mathbb{R}$	nonlinear, Lipschitz continuous function.
$\Theta : \mathbb{R}^{f_0} \to \mathbb{R}^{f_L}$	Simplified Graph Convolution network without output nonlinearity
$\hat{\Theta} : \mathbb{R}^{f_0} \to \mathbb{R}^{f_L}$	Simplified Graph Convolution network with output nonlinearity
$\boldsymbol{\xi} \in \mathbb{R}^f$	f-dimensional random variable.
$\boldsymbol{\xi}_i^{(l)} \in \mathbb{R}^{f_l}$	random variable representing random features of node $v_i \in V$ at layer l
$\vec{\boldsymbol{\xi}}^{(l)} \in \mathbb{R}^{nf_l}$	random vector resulting from concatenation of the features of all nodes
$\Xi^{(l)} \in \mathbb{R}^{n \times f_l}$	random matrix representing random node features at layer l.
$\mathbb{P}_{\boldsymbol{\xi}}$	distribution (or law) of the random variable $\boldsymbol{\xi}$
$p_{\boldsymbol{\xi}}$	probability density function of the random variable $\boldsymbol{\xi}$
$\mathcal{B}(\mathbb{R}^f)$	the Borel sets of $\mathbb{R}^f$
$\Psi_\sharp \mathbb{P}$	pushforward of the distribution $\mathbb{P}$ under the function Ψ
$W_p(\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}})$	p-Wasserstein distance between the probability distributions $\mathbb{P}_{\boldsymbol{\xi}}$ and $\mathbb{P}_{\boldsymbol{\xi'}}$
$\Gamma(\mathbb{P}_\xi, \mathbb{P}_{\xi'})$	space of couplings of the random variables ξ and ξ'
$\gamma \in \Gamma(\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}})$	coupling between the random variables $\boldsymbol{\xi}$ and $\boldsymbol{\xi'}$
$\mu_{\boldsymbol{\xi}}$	first moment of the random variable $\boldsymbol{\xi}$
$\Sigma_{\boldsymbol{\xi}}$	second moment of the random variable $\boldsymbol{\xi}$
FCD_p	Wasserstein distance-based node level pseudometric
$\|\| \cdot \|\|_p$	L_p-norm
L_p	space of functions with integrable p norm
$\mathcal{N}(\mu, \Sigma)$	the (multivariate) normal distribution with mean $\mu \in \mathbb{R}^f$ and covariance $\Sigma \in \mathbb{R}^{f \times f}$
C	Lipschitz constant of a GCN with respect to the L_2 norm
C_L	Lipschitz constant of SGC with respect to the FCD_p distance.
L	Lower-triangular matrix in Cholesky decomposition
$\Phi_k, k \in \{1, \ldots, K\}$	Orthonormal polynomial basis function
$\boldsymbol{c}_k$	PCE coefficients

Table 1: Notation used throughout the paper.

A.1 Graphs

A **graph** $G = (V, E)$ is defined by a set of nodes V and the edges $E \subset V \times V$ between them. We denote the number of nodes $n = |V|$ and the number of edges $m = |E|$. A graph can be represented by its **adjacency matrix** $A \in \mathbb{R}^{n \times n}$, where $A_{ij} = 1$ if node v_i is connected to node v_j (i.e, $(v_i, v_j) \in E$), and $A_{ij} = 0$ otherwise. A node $v_i \in V$ often possesses additional properties

encoded in its **feature vector** $\boldsymbol{x}_i \in \mathbb{R}^{1\times f}$, where f is called the feature dimension. It is convenient to gather the feature vectors of all nodes in a feature matrix $X \in \mathbb{R}^{n\times f}$. For example, to model a social media platform as a graph, one represents each user as a node, their characteristics (e.g., age and topics of interests) as node features, and which users are friends on the platform with edges. The degree $d_i = \sum_{j=1}^{n} A_{ij}$ of a node is the number of other nodes that it is connected to. We denote the diagonal matrix of degrees as $D = \mathrm{diag}(d_1, \ldots, d_n)$.

A.2 Functions of Random Variables

The **pushforward** is a way to calculate how uncertainty in the inputs of a function translates into uncertainty of its outputs. Let $\boldsymbol{\xi}$ be a multivariate random variable taking values in $\mathbb{R}^{f_1}$, e.g., the input to a neural network, and $\Psi : \mathbb{R}^{f_1} \to \mathbb{R}^{f_2}$ a Borel function, e.g., a neural network. Applying Ψ to $\boldsymbol{\xi}$ defines another random variable $\boldsymbol{\eta} = \Psi(\boldsymbol{\xi})$ taking values in $\mathbb{R}^{f_2}$ and describing in our examples the output of a neural network. When $\boldsymbol{\xi}$ has distribution $\mathbb{P}_{\boldsymbol{\xi}}$, the distribution $\mathbb{P}_{\boldsymbol{\eta}}$ of $\boldsymbol{\eta}$ is

$$\mathbb{P}_{\boldsymbol{\eta}}(\boldsymbol{\eta} \in B) = \mathbb{P}_{\boldsymbol{\eta}}(\Psi(\boldsymbol{\xi}) \in B) = \mathbb{P}_{\boldsymbol{\xi}}(\boldsymbol{\xi} \in \Psi^{-1}(B)) \;\; \forall B \in \mathcal{B}(\mathbb{R}^{f_2}), \tag{24}$$

where $\mathcal{B}(\mathbb{R}^{f_2})$ are the Borel sets of $\mathbb{R}^{f_2}$ and $\Psi^{-1}(B) \in \mathcal{B}(\mathbb{R}^{f_1})$ is the preimage of $B \in \mathcal{B}(\mathbb{R}^{f_2})$ under Ψ, a Borel set of $\mathbb{R}^{f_1}$.

A.3 Couplings of Random Variables

Given two random variables $\boldsymbol{\xi} \in \mathbb{R}^{f}$ and $\boldsymbol{\xi'} \in \mathbb{R}^{f'}$ with distributions $\mathbb{P}_{\boldsymbol{\xi}}$ and $\mathbb{P}_{\boldsymbol{\xi'}}$ respectively, a **coupling** is a distribution γ on the product space $\mathbb{R}^{f} \times \mathbb{R}^{f'}$ that has $\mathbb{P}_{\boldsymbol{\xi}}$ and $\mathbb{P}_{\boldsymbol{\xi'}}$ as its marginals, i.e.,

$$\gamma(B \times \mathbb{R}^{f'}) = \mathbb{P}_{\boldsymbol{\xi}}(B) \text{ and } \gamma(\mathbb{R}^{f} \times B') = \mathbb{P}_{\boldsymbol{\xi'}}(B'), \tag{25}$$

for all $B \in \mathcal{B}(\mathbb{R}^{f})$ and $B' \in \mathcal{B}(\mathbb{R}^{f'})$. We denote the space of all couplings as $\Gamma(\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}})$. Sometimes the word coupling is also used for a random variable that has distribution γ.

As for any other random variable, one can compute the **pushforward of a coupling**. Here we are interested in a special case, where $\boldsymbol{\xi}, \boldsymbol{\xi'} \in \mathbb{R}^{f}$ be random variables with distributions $\mathbb{P}_{\boldsymbol{\xi}}, \mathbb{P}_{\boldsymbol{\xi'}}$ with a coupling γ between them, and the same Borel function $\Psi : \mathbb{R}^{f} \to \mathbb{R}^{f'}$ is applied to each of them individually, in this case the pushforward of the coupling is

$$(\Psi \times \Psi)_{\sharp\gamma}(B \times B') = \gamma(\Psi^{-1}(B) \times \Psi^{-1}(B')), \tag{26}$$

for all Borel sets $B, B' \in \mathcal{B}(\mathbb{R}^{f})$.

B Polynomial Chaos Expansion

Here, we describe how to calculate the first moments of multivariate random variables associated with nodes after their transformation by a nonlinear function using PCE. We assume that the random variables at the graph nodes form a joint distribution with dependent marginals, reflecting realistic dependencies in graph-structured data such as those arising from message-passing operations in MPNNs.

We address the case where the joint distribution is multivariate Gaussian with dependent marginals – i.e., $\boldsymbol{\xi} \sim \mathcal{N}(\mu, \Sigma) \in \mathbb{R}^{f}$. First, we eliminate dependencies with the Cholesky decomposition of the covariance matrix Σ as follows:

$$\Sigma = LL^{T}, \tag{27}$$

where L is lower-triangular matrix. Using this decomposition, one can represent $\boldsymbol{\xi}$ as:

$$\boldsymbol{\xi} = \mu + L\boldsymbol{z} \tag{28}$$

where $\boldsymbol{z} \sim \mathcal{N}(0, I)$. This allows us to use standard multivariate Hermite polynomials as the polynomial basis, which are orthogonal w.r.t. the independent Gaussian measure [Schnabel, 2023].

We need to calculate the moments of the nonlinearity $\sigma(\cdot)$ applied elementwise to the random variables in $\boldsymbol{\xi}$ which has the following representation:

$$\sigma(\boldsymbol{\xi}_i) = \sigma_i(\boldsymbol{z}) = \sigma(\mu_i + l_i^T \boldsymbol{z}), \tag{29}$$

where l_i is the i_{th} row of the matrix L. In order to use PCE, nonlinearity $\sigma(\cdot)$ shall satisfy:

$$\sigma(\boldsymbol{\xi}_i) \in L^2(p_{\boldsymbol{z}}) \Longleftrightarrow \int_{\mathbb{R}^f} \left|\sigma(\mu_i + l_i^T \boldsymbol{z})\right|^2 dp_{\boldsymbol{z}}(\boldsymbol{z}) < \infty. \tag{30}$$

All common nonlinear functions in MPNNs satisfy these conditions w.r.t Gaussian measure. Therefore, we are considering the expansion of the random vector:

$$\sigma(\boldsymbol{\xi}) = (\sigma_1(\boldsymbol{z}), \cdots, \sigma_f(\boldsymbol{z})) \tag{31}$$

which according to [Sullivan, 2015] can be expanded as:

$$\sigma(\boldsymbol{\xi}) = \sum_{k=0}^{K} \boldsymbol{c}_k \Phi_k(\boldsymbol{z}) \tag{32}$$

with vector-valued coefficients $\boldsymbol{c}_k = [c_{1k}, \ldots, c_{fk}]$ for each k, where $\Phi_k(\boldsymbol{z})$ are basis functions from a multivariate orthonormal polynomial basis with respect to the probability distribution $p_{\boldsymbol{z}}$ defined on the input space $\mathcal{D}_{\boldsymbol{z}}$. In the case of Gaussian random variables, the natural choice for $\Phi_k(\boldsymbol{z})$ are multivariate Hermite polynomials. The mean and covariance have the following representation:

$$\mu_{\sigma(\boldsymbol{\xi})} = \boldsymbol{c}_0, \quad \Sigma_{\sigma(\boldsymbol{\xi})} = \sum_{k=1}^{K} \boldsymbol{c}_k \boldsymbol{c}_k^T, \tag{33}$$

where

$$[\Sigma_{\sigma(\boldsymbol{\xi})}]_{ij} = \sum_{k=1}^{K} c_{i,k}\, c_{j,k} \tag{34}$$

and

$$c_{ik} = \int_{\mathbb{R}^f} \sigma_i(\boldsymbol{z}) \Phi_k(\boldsymbol{z}) dp_{\boldsymbol{z}}(\boldsymbol{z}). \tag{35}$$

The coefficients c_{ik} can be obtained with Gaussian quadrature or Smolyak's sparse quadrature techniques [Schnabel, 2023].

C Certified Robustness

Theorem 4. *MPNN for node classification tasks is robust against L_2-norm feature perturbation $||\Delta|| = \epsilon$ with probability $1 - \delta$ if:*

$$\epsilon < \min_{y \neq y^\star} \frac{\hat{\mu}_{\boldsymbol{\xi}^z_{iy}} - \sqrt{\hat{\Sigma}_{\boldsymbol{\xi}^z_{iy}}} \sqrt{\frac{1-\delta_y}{\delta_y}}}{\sqrt{2}C}, \tag{36}$$

where $\delta = \sum_y \delta_y$, δ_y- probability of misclassification to class y, $\hat{\mu}_{\boldsymbol{\xi}^z_{iy}} = \mu_{\boldsymbol{\xi}^z_{i,y^\star}} - \mu_{\boldsymbol{\xi}^z_{i,y}}$ is the mean and $\hat{\Sigma}_{\boldsymbol{\xi}^z_{iy}} = \Sigma_{\boldsymbol{\xi}^z_i, y^\star y^\star} + \Sigma_{\boldsymbol{\xi}^z_i, yy} - 2\,\Sigma_{\boldsymbol{\xi}^z_i, y^\star y}$ the covariance of the random margin between the element

of the logit associated with true label class $y^\star$ and logit element associated with any other class label $y \neq y^\star$, C is the Lipschitz constant of the GCN with respect to the L_2-norm node feature perturbation.

Proof. Let us consider that every node v_i has a true classification label $y_i^\star \in \{1, \cdots, f_L\}$. For $\forall\, y \neq y^\star$, the margin in logits is:

$$M_y^z = \boldsymbol{\xi}_{iy^\star}^z - \boldsymbol{\xi}_{iy}^z = (\boldsymbol{e}_{y^\star} - \boldsymbol{e}_y)^\mathsf{T} \boldsymbol{\xi}_i^z = \boldsymbol{m}_y^\mathsf{T} \boldsymbol{\xi}_i^z, \tag{37}$$

where $\boldsymbol{e}_y \in \mathbb{R}^{f_L}$ is a standard basis vector, and $\boldsymbol{m}_y = \boldsymbol{e}_{y^\star} - \boldsymbol{e}_y$

Let us now consider the case when the perturbation $||\Delta|| = \epsilon$ was added to the node feature vector $\boldsymbol{\xi}_i^{(0)}$, we define the corresponding logit as $\boldsymbol{\xi}_i^{z'}$. Let us consider the difference between the logit margins of the original and perturbed node features:

$$\begin{aligned} \left| M_y^z - M_y^{z'} \right| = \left| \boldsymbol{m}_y^\mathsf{T} (\boldsymbol{\xi}_i^z - \boldsymbol{\xi}_i^{z'}) \right| \leq \left|\left| \boldsymbol{m}_y^\mathsf{T} \right|\right| \left|\left| \boldsymbol{\xi}_i^z - \boldsymbol{\xi}_i^{z'} \right|\right| = \sqrt{2} \left|\left| \boldsymbol{\xi}_i^z - \boldsymbol{\xi}_i^{z'} \right|\right| \leq \\ \sqrt{2} C \left|\left| \boldsymbol{\xi}_i^{(0)} - \boldsymbol{\xi}_i^{'(0)} \right|\right| = \sqrt{2} C \left|\left| \Delta \right|\right| = \sqrt{2} C \epsilon, \end{aligned} \tag{38}$$

or

$$M_y^z - \sqrt{2} C \epsilon, \leq M_y^{z'} \leq M_y^z + \sqrt{2} C \epsilon, , \tag{39}$$

where C is the L_2 Lipschitz constant of GCN.

Let us now consider the probability of misclassification $\mathbb{P}(M_y^{z'} \leq 0)$. From the previous expression, it follows that $\mathbb{P}[M_y^{z'} \leq 0] \leq \mathbb{P}[M_y^z \leq \sqrt{2} C \epsilon]$. By leveraging Cantelli inequality we can derive an upper bound on the misclassification probability:

$$\mathbb{P}[M_y^{z'} \leq 0] \leq \mathbb{P}[M_y^z \leq \sqrt{2} C \epsilon] \leq \frac{\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z}}{\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z} + (\hat{\mu}_{\boldsymbol{\xi}_{iy}^z} - \sqrt{2} C \epsilon)^2}, \tag{40}$$

where $\hat{\mu}_{\boldsymbol{\xi}_{iy}^z} = \mu_{\boldsymbol{\xi}_{i,y^\star}^z} - \mu_{\boldsymbol{\xi}_{i,y}^z}$, $\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z} = \Sigma_{\boldsymbol{\xi}_i^z, y^\star y^\star} + \Sigma_{\boldsymbol{\xi}_i^z, yy} - 2\, \Sigma_{\boldsymbol{\xi}_i^z, y^\star y}$.

Setting the upper bound on the probability of misclassification to be δ_y, and solving for ϵ, we get:

$$\epsilon = \frac{\hat{\mu}_{\boldsymbol{\xi}_{iy}^z} - \sqrt{\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z}} \sqrt{\frac{1-\delta_y}{\delta_y}}}{\sqrt{2} C} \tag{41}$$

Each class has its own ϵ, so in order to be robust against misclassification to any label, we say that MPNN is robust for node classification task with probability at least $1 - \delta$ if:

$$\epsilon < \min_{y \neq y^\star} \frac{\hat{\mu}_{\boldsymbol{\xi}_{iy}^z} - \sqrt{\hat{\Sigma}_{\boldsymbol{\xi}_{iy}^z}} \sqrt{\frac{1-\delta_y}{\delta_y}}}{\sqrt{2} C}, \tag{42}$$

where $\delta = \sum_y \delta_y$. □

D Discriminative Power

Theorem 5. FCD_p *has the same discriminative power as* $\hat{\Theta}(\cdot)$*:*

$$W_p(\hat{\Theta}(v_i), \hat{\Theta}(v_j)) > 0 \Rightarrow \mathrm{FCD}_p(v_i, v_j) > 0 \tag{43}$$

Proof. Let $\Psi : \mathbb{R}^{f_0} \to \mathbb{R}^{f_L}$ be the application of weights and element-wise nonlinearity to the node feature matrix $X^s = S^L X^{(0)}$ after L-layer structural update of SGC, $\Psi(X^s) = \sigma(X^s W)$. In order to prove the discriminative power we need to show that

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) > 0 \Rightarrow W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) > 0, \tag{44}$$

where $\boldsymbol{\xi}_i^s = [S^L \Xi]_i$ and $\boldsymbol{\xi}_j^s = [S^L \Xi]_j$ are the node features after structural update in the L layer SGC architecture and defined in terms of the matrix valued random variable $\Xi \in \mathbb{R}^{n \times f_0}$ of input node features. Eq. (44)$\Rightarrow$ If

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) > 0, \tag{45}$$

then

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \neq \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}. \tag{46}$$

Therefore, $\exists B \in \mathcal{B}(\mathbb{R}^{f_L})$, such that

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}(B) \neq \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}(B) \tag{47}$$

or by definition

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(\Psi^{-1}(B)) \neq \mathbb{P}_{\boldsymbol{\xi}_j^s}(\Psi^{-1}(B)). \tag{48}$$

From the definition of pushforward measure, $\exists B' \in \mathcal{B}(\mathbb{R}^{f_0})$, such that

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(B') \neq \mathbb{P}_{\boldsymbol{\xi}_j^s}(B'). \tag{49}$$

Therefore,

$$\mathbb{P}_{\boldsymbol{\xi}_i^s} \neq \mathbb{P}_{\boldsymbol{\xi}_j^s} \tag{50}$$

and

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = \mathrm{FCD}_p(v_i, v_j) > 0. \tag{51}$$

□

Theorem 6. FCD_p *has the following discriminative power w.r.t.* $\hat{\Theta}$ *when the matrix of weights* W *and* σ *are invertible.*

$$\mathrm{FCD}_p(v_i, v_j) = 0 \Leftrightarrow W_p(\hat{\Theta}(v_i), \hat{\Theta}(v_j)) = 0 \tag{52}$$

$$\mathrm{FCD}_p(v_i, v_j) > 0 \Leftrightarrow W_p(\hat{\Theta}(v_i), \hat{\Theta}(v_j)) > 0 \tag{53}$$

Proof. Let $\Psi : \mathbb{R}^{f_0} \to \mathbb{R}^{f_L}$ be the application of weights and element-wise nonlinearity to the node feature matrix Ξ^s after structural update, defined as $\Psi(X^s) = \sigma(X^s W)$. If W and σ are invertible then Ψ is a bijective continuous function with continuous inverse $\Psi^{-1}(\cdot)$, therefore $\Psi(\cdot)$ is homeomorphism. Note that this also means that $f_0 = f_L = f$. In order to prove the discriminative power we need to show that

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = 0 \Leftrightarrow W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) = 0 \tag{54}$$

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) > 0 \Leftrightarrow W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) > 0 \tag{55}$$

Eq. (54) $\Rightarrow$: If

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = 0, \tag{56}$$

then from the properties of Wasserstein distance which is a true metric

$$\mathbb{P}_{\boldsymbol{\xi}_i^s} = \mathbb{P}_{\boldsymbol{\xi}_j^s}, \tag{57}$$

which means that $\forall B \in \mathcal{B}(\mathbb{R}^f)$

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(B) = \mathbb{P}_{\boldsymbol{\xi}_j^s}(B). \tag{58}$$

By definition of pullback measure, and because Ψ is a homeomorphism, this is equivalent to

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(\Psi^{-1}(B)) = \mathbb{P}_{\boldsymbol{\xi}_j^s}(\Psi^{-1}(B)),\ \forall B \in \mathcal{B}(\mathbb{R}^f), \tag{59}$$

which by definition means that

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}(B) = \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}(B),\ \forall B \in \mathcal{B}(\mathbb{R}^f), \tag{60}$$

and therefore

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} = \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}. \tag{61}$$

From the properties of Wasserstein distance we have that

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) = \mathrm{FCD}_p(v_i, v_j) = 0. \tag{62}$$

Eq. (54) $\Leftarrow$: If

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) = 0, \tag{63}$$

then from the properties of Wasserstein distance, which is a true metric the following holds

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} = \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}} \tag{64}$$

or equivalently

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}(B) = \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}(B),\ \forall B \in \mathcal{B}(\mathbb{R}^{f_L}), \tag{65}$$

which by definition means that

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(\Psi^{-1}(B)) = \mathbb{P}_{\boldsymbol{\xi}_j^s}(\Psi^{-1}(B)),\ \forall B \in \mathcal{B}(\mathbb{R}^{f_L}). \tag{66}$$

By definition of pushforward measure and because Ψ is a homeomorphism, this means

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(B') = \mathbb{P}_{\boldsymbol{\xi}_j^s}(B'),\ \forall B' \in \mathcal{B}(\mathbb{R}^{f_0}) \tag{67}$$

or

$$\mathbb{P}_{\boldsymbol{\xi}_i^s} = \mathbb{P}_{\boldsymbol{\xi}_j^s} \tag{68}$$

and thus

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}\mathbb{P}_{\boldsymbol{\xi}_j^s}) = \mathrm{FCD}_p(v_i, v_j) = 0. \tag{69}$$

Eq. (55)$\Rightarrow$: If

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) > 0 \tag{70}$$

then

$$\mathbb{P}_{\boldsymbol{\xi}_i^s} \neq \mathbb{P}_{\boldsymbol{\xi}_j^s}. \tag{71}$$

Therefore, $\exists B' \in \mathcal{B}(\mathbb{R}^{f_0})$, such that

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(B') \neq \mathbb{P}_{\boldsymbol{\xi}_j^s}(B'). \tag{72}$$

By the definition of pullback measure, $\exists B \in \mathcal{B}(\mathbb{R}^{f_L})$ such that

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}(\Psi^{-1}(B)) \neq \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}(\Psi^{-1}(B)), \tag{73}$$

thus

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \neq \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}} \tag{74}$$

and

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) = \mathrm{FCD}_p(v_i, v_j) > 0. \tag{75}$$

Eq. (55)$\Leftarrow$: If

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) > 0, \tag{76}$$

then

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}} \neq \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}. \tag{77}$$

Therefore, $\exists B \in \mathcal{B}(\mathbb{R}^{f_L})$, such that

$$\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}(B) \neq \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}(B) \tag{78}$$

or by definition

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(\Psi^{-1}(B)) \neq \mathbb{P}_{\boldsymbol{\xi}_j^s}(\Psi^{-1}(B)) \tag{79}$$

From the definition of pushforward measure, $\exists B' \in \mathcal{B}(\mathbb{R}^{f_0})$, such that

$$\mathbb{P}_{\boldsymbol{\xi}_i^s}(B') \neq \mathbb{P}_{\boldsymbol{\xi}_j^s}(B'). \tag{80}$$

Therefore, it holds that

$$\mathbb{P}_{\boldsymbol{\xi}_i^s} \neq \mathbb{P}_{\boldsymbol{\xi}_j^s} \tag{81}$$

and

$$W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = \mathrm{FCD}_p(v_i, v_j) > 0. \tag{82}$$

□

Corollary 4. *The discriminative power of* FCD_p *is the same as for* $\hat{\Theta}(\cdot)$ *in the default SGC architecture* $\Theta(\cdot)$.

E Lipschitz Continuity of SGC with Nonlinear Activation

Theorem 7. $\hat{\Theta}(\cdot)$ *is a globally Lipschitz function w.r.t.* FCD_p*:*

$$W_p(\hat{\Theta}(v_i), \hat{\Theta}(v_j)) \leq C_L \cdot \mathrm{FCD}_p(v_i, v_j) \tag{83}$$

Proof. We need to show that

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) \leq C_L W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}). \tag{84}$$

Let $\Psi(X^s) = \sigma(X^s W)$ be the application of weights and nonlinearity after the structural update. The function $\Psi : \mathbb{R}^f \to \mathbb{R}^f$ is Lipschitz continuous in L_p-norm if σ is Lipschitz continuous (in L_p -norm) and we denote its Lipschitz constant as C_L,

$$||\Psi(x) - \Psi(y)||_2 \leq C_L \, ||x - y||_2 \, . \tag{85}$$

Let $\boldsymbol{\xi}_i^s, \boldsymbol{\xi}_j^s \in \mathbb{R}^f$ be the random variables of node features after structural updates at nodes v_i and v_j respectively. Let $\Gamma(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s})$ the space of couplings between them, and let $\Gamma(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}})$ be the space of couplings after pushforward through Ψ. The elements γ' of $\Gamma(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}})$ are of the form $\gamma' = (\Psi \times \Psi)_{\sharp\gamma}$.

For all $A \in \mathcal{B}(\mathbb{R}^f)$ we have:

$$\begin{aligned} \gamma'(A \times \mathbb{R}^f) = (\Psi \times \Psi)_\sharp \gamma(A \times \mathbb{R}^f) = \gamma((\Psi \times \Psi)^{-1}(A \times \mathbb{R}^f)) = \\ = \gamma(\Psi^{-1}(A) \times \mathbb{R}^f) = \mathbb{P}_{\boldsymbol{\xi}_i^s}(\Psi^{-1}(A)) = \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}(A). \end{aligned} \tag{86}$$

And mutatis mutandis for the other variable,

$$\gamma'(\mathbb{R}^f \times B) = \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}(B). \tag{87}$$

The p-Wasserstein distance between the two pushforward probability measures is given by:

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) = \left(\inf_{\gamma' \in \Gamma(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}})} \iint_{\mathbb{R}^f \times \mathbb{R}^f} \|u - v\|^p \, d\gamma'(u, v) \right)^{1/p}. \tag{88}$$

Using Eq. (85), it is clear that

$$\begin{aligned} \iint_{\mathbb{R}^f \times \mathbb{R}^f} \|u - v\|^p \, d\gamma'(u, v) = \iint_{\mathbb{R}^f \times \mathbb{R}^f} \|\Psi(x) - \Psi(y)\|^p \, d\gamma(x, y) \leq \\ \leq C_L^p \iint_{\mathbb{R}^f \times \mathbb{R}^f} \|x - y\|^p \, d\gamma(x, y). \end{aligned} \tag{89}$$

Since this holds for any coupling $\gamma \in \Gamma(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s})$, it holds for the infimum over such couplings

$$\begin{aligned} W_p^p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) &= \inf_{\gamma' \in \Gamma(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}})} \iint_{\mathbb{R}^f \times \mathbb{R}^f} \|u - v\|^p \, d\gamma'(u, v) \leq \\ &\leq C_L^p \inf_{\gamma \in \Gamma(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s})} \iint_{\mathbb{R}^f \times \mathbb{R}^f} \|x - y\|^p \, d\gamma(x, y) = C_L^p W_p^p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}). \end{aligned} \tag{90}$$

Taking the p-th root on both sides yields the final bound

$$W_p(\Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_i^s}}, \Psi_{\sharp\mathbb{P}_{\boldsymbol{\xi}_j^s}}) \leq C_L W_p(\mathbb{P}_{\boldsymbol{\xi}_i^s}, \mathbb{P}_{\boldsymbol{\xi}_j^s}) = C_L \mathrm{FCD}_p(v_i, v_j). \tag{91}$$

□

Corollary 5. *The Lipschitz continuity remains valid in the default SGC architecture* $\Theta(\cdot)$.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes] .

   Justification: The abstract and introduction accurately summarize the paper's core contributions, including uncertainty quantification in message passing neural networks, robustness guarantees against L_2-norm feature perturbations, and the introduction of a novel pseudometric to study generalization guarantees, while rigorously defining all assumptions and limitations of the proposed methodology.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes] .

   Justification: The paper acknowledges several limitations: reliance on Gaussian input assumptions and the focus on the SGC architecture in the introduction of the novel pseudometric.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes] .

   Justification: The paper provides all assumptions in the theorems formulations. Complete and correct proofs of the theorems are provided in Appendix.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes] .

   Justification: The paper fully discloses the information needed to reproduce the main experimental results, including dataset details, model architectures, training setups, and evaluation methodologies.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No] .

   Justification: The paper does not provide open access to the code for experimental results.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes] .

   Justification: The paper specifies the training and test details, the choice of hyperparameters, type of optimizer and number of epochs used in training.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA] .

   Justification: There is no way to measure the statistical significance of the experiments in the paper.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No] .

   Justification: The computational resources needed to run experiments are minimal, therefore, are not mentioned in the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes] .

   Justification: The research conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA] .

Justification: The paper does not have any potential societal impacts.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA] .

Justification: The paper does not introduce any data or models that have a high risk of misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes] .

Justification: The owners of datasets used for training the models are properly credited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA] .

    Justification: The methodology does not introduce any new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA] .

    Justification: The methodology does not include any studies with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA] .

    Justification: The methodology does not include any studies with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [No] .

    Justification: The core methodology does not involve LLMs as important component.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.