ON THE SAMPLE COMPLEXITY OF GNNS

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph Neural Networks (GNNs) have demonstrated strong empirical performance across domains, yet their fundamental statistical behavior remains poorly understood. This paper presents a theoretical characterization of the sample complexity of ReLU-based GNNs. We establish tight minimax lower bounds on the generalization error, showing that for arbitrary graphs, *without structural assumptions* (i.e., in the worst case over admissible graphs), it scales as $\sqrt{\frac{\log d}{n}}$ with sample size n and input dimension d, matching the $1/\sqrt{n}$ behavior known for feed-forward neural networks. Under structural graph assumptions—specifically, strong homophily and bounded spectral expansion—we derive a sharper lower bound of $\frac{d}{\log n}$. Empirical results on standard datasets (Cora, Reddit, QM9, Facebook) using GCN, GAT, and GraphSAGE support these theoretical predictions. Our findings establish fundamental limits on GNN generalization and underscore the role of graph structure in determining sample efficiency.

1 INTRODUCTION

Graph Neural Networks (GNNs) have become central to machine learning on structured data, achieving state-of-the-art results across domains such as social networks (Sen et al., 2008), molecular property prediction (Ruddigkeit et al., 2012), and community detection (Ramakrishnan et al., 2014a). By exploiting graph topology and node features, GNNs are now indispensable in modern AI systems. Despite this success, their statistical foundations remain limited: *how many training samples are required for a GNN to generalize reliably to unseen data?*

For feed-forward and convolutional networks, minimax analyses show that ReLU networks achieve risk scaling of $1/\sqrt{n}$ in the number of samples n (Golestaneh et al., 2024), in contrast to the classical $1/n$ parametric rate. These results rely on i.i.d. assumptions, whereas GNNs operate on correlated inputs through graph edges. This dependency complicates sample complexity analysis and raises the central question: *how does graph structure influence generalization?*

While prior work has provided *upper bounds* for GNNs using VC-dimension or PAC-Bayes frameworks, these bounds scale poorly with network size and give limited insight into fundamental limits (see Section 2). In particular, sharp *lower bounds* for GNNs are largely absent, leaving it unclear whether GNNs can match the sample efficiency of feed-forward networks, or whether structural biases induce fundamentally different scaling laws.

In this work, we establish new minimax lower bounds for ReLU-based GNNs. Using Fano's inequality, we prove that *without structural assumptions on the graph* (worst case over admissible graphs), the generalization error must scale at least as $\sqrt{\frac{\log d}{n}}$, matching the known $1/\sqrt{n}$ rate. Moreover, under natural conditions—graphs with strong homophily and moderate expansion (Laplacian spectral gap $\lambda_2 \leq \kappa/\log n$)—we obtain a sharper lower bound of $\frac{d}{\log n}$.

Experiments on Cora (node classification), Reddit (community detection), QM9 (graph regression), and Facebook (link prediction) with standard GNN architectures (GCN (Kipf & Welling, 2017), GAT (Veličković et al., 2018), GraphSAGE (Hamilton et al., 2017)) confirm that generalization often aligns with this refined $1/\log n$ scaling.

Contributions. We:

1. Establish a minimax risk lower bound for GNNs, scaling as $\sqrt{\log d/n}$.

2. Derive a sharper lower bound of $\frac{d}{\log n}$ under structural graph assumptions.
3. Empirically validate this refined scaling law on four benchmark datasets and three widely adopted GNN architectures (GCN, GAT, GraphSAGE).
4. Provide a framework connecting theoretical sample complexity with practical GNN performance.

2 Related Work

The sample complexity of deep neural networks is well studied. For fully connected and convolutional architectures, the minimax risk is known to scale as $1/\sqrt{n}$, reflecting the higher data requirements of deep learning models compared to classical parametric methods (Golestaneh et al., 2024). Nonparametric regression under smoothness assumptions also yields convergence guarantees (Schmidt-Hieber, 2020), though these results differ substantially from those for modern deep architectures.

In contrast, the theoretical understanding of generalization in Graph Neural Networks (GNNs) remains underdeveloped. Early efforts analyzed the VC-dimension of GNNs (Scarselli et al., 2009), but obtained bounds that scale poorly with depth and width. PAC-Bayesian approaches provided stability-based alternatives (Liao et al., 2020), yet sharp sample complexity characterizations are still lacking. Other lines of work investigate representational limits (Garg et al., 2020), or connect graph topology to training dynamics (Oono & Suzuki, 2021; Nikolentzos et al., 2022). However, lower bounds on generalization—critical for understanding statistical limitations—remain scarce.

Expressivity and generalization of MPNNs. Franks et al. study message-passing GNNs from an expressivity–learnability perspective, establishing *upper* generalization bounds via VC/covering-number analyses and showing how node individualization or positional encodings increase expressivity while preserving learnability (Franks et al., 2024). Their guarantees scale with architectural size (depth/width) and the chosen individualization scheme. Our results are complementary: we provide *minimax lower bounds* for standard ReLU MPNNs with input-independent local aggregation (Assumption (A1)), making the role of graph structure explicit via the spectral–homophily condition (Theorem 2). In short, Franks et al. (2024) delineate what is achievable in favorable regimes (upper bounds), whereas our results certify obstacles that persist even for richer hypothesis classes (by monotonicity of minimax risk).

Recently, Pellizzoni et al. analyzed GNNs with node individualization schemes, showing that such modifications reduce sample complexity by enhancing expressivity while controlling VC-dimension and covering numbers (Pellizzoni et al., 2024). Together with Franks et al. (2024), these works chart the *upper-bound* landscape under expressivity-enhancing augmentations (e.g., individualization or positional encodings). Our focus is orthogonal: we establish *lower* bounds for standard message-passing GNNs without such augmentations, exposing an unavoidable dependence on graph structure.

Our work extends the minimax framework from feedforward networks (Golestaneh et al., 2024) to GNNs with arbitrary graph inputs, without relying on strong smoothness or independence assumptions. By incorporating graph topology directly, we derive intrinsic lower bounds on GNN sample complexity that align closely with empirical trends. Unlike our general bound (Theorem 1), the structure-aware bound (Theorem 2) accommodates adjacency-masked attention by relying on mixing/locality rather than input-independent aggregation.

Taken together, these strands bracket the problem: expressivity-driven *upper* bounds (Pellizzoni et al., 2024; Franks et al., 2024) and structure-aware *lower* bounds (this work).

3 Problem Formulation and Main Result

We consider a GNN operating on a graph $G = (V, E)$ with $|V|$ nodes, $|E|$ edges, adjacency matrix A, and node features $X_v \in \mathbb{R}^{|V| \times d}$ for $v \in V$.

Graphs and terminology. Throughout, we allow arbitrary simple, undirected graphs. A *chain graph* (path graph P_m on m nodes) has edges $\{(1,2),(2,3),\ldots,(m-1,m)\}$. Chain graphs are admissible members of our graph family and instantiate the hard distribution in the proof of Theorem 1.

Task settings. We study three prediction regimes with $\hat{Y}$ the output of a GNN f, and $q \geq 1$ its output dimension: (i) *Graph-level (inductive)*: Each example is a graph G with features X, and the

model outputs $f(G, X) = \hat{Y} \in \mathbb{R}^q$. (ii) *Node-level (transductive)*: A single graph G is observed; training/test examples are nodes $v \in V$. The model outputs $f(G, X) = \hat{Y} \in \mathbb{R}^{|V| \times q}$, with the v-th row $\hat{y}_v$ predicting node v. (iii) *Link-level*: Given queried pairs $\mathcal{P} \subseteq V \times V$, the model outputs $f(G, X; \mathcal{P}) = \hat{Y} \in \mathbb{R}^{|\mathcal{P}| \times q}$, with entries $\hat{y}_{(u,v)}$ from final-layer embeddings.

Unless stated otherwise, losses are squared error for regression and cross-entropy for classification. Theorem 1 concerns graph-level (inductive) risk, and Theorem 2 node-level (transductive) risk.

ReLU Graph Neural Networks. A ReLU-based GNN with L message-passing layers realizes a function $f \colon G \mapsto \hat{Y}$, where G is a graph with node features X, and $\hat{Y}$ is the predicted output. Each layer updates hidden node representations as:

$$h_i^{(\ell+1)} = \phi\left(W^{(\ell)} \operatorname{Agg}_{j \in \mathcal{N}(i)} h_j^{(\ell)} + B^{(\ell)} h_i^{(\ell)}\right), \quad \phi(z) = \max\{0, z\}, \quad \ell = 0, \ldots, L-1. \tag{1}$$

Here $W^{(\ell)} \in \mathbb{R}^{d_{\ell+1} \times d_\ell}$ acts on the aggregated neighbor messages $\operatorname{Agg}_{j \in \mathcal{N}(i)} h_j^{(\ell)}$, and $B^{(\ell)} \in \mathbb{R}^{d_{\ell+1} \times d_\ell}$ is the self–loop mixing matrix applied to $h_i^{(\ell)}$. Additive biases $b^{(\ell)} \in \mathbb{R}^{d_{\ell+1}}$ can be included but are omitted here since including them only enlarges the hypothesis class and does not affect our minimax lower bounds. The aggregator Agg is permutation-invariant, graph-dependent but input-independent (e.g., *sum* or *mean*); node representations are initialized as $h_i^{(0)} = x_i$

Architectural scope and assumptions. Our lower bound in Theorem 1 applies to message-passing GNNs that satisfy: **(A1)** input-independent, 1-hop permutation-invariant aggregation (e.g., SUM, MEAN, normalized adjacency), and **(A2)** uniform layerwise Lipschitz/variation control, instantiated as the ℓ_1-norm budget $\sum_{\ell=0}^{L-1} \left(\|W^{(\ell)}\|_1 + \|B^{(\ell)}\|_1\right) \le v_s$, which promotes sparsity and is consistent with recent theoretical results on over-parameterized networks (Lederer, 2022; Taheri et al., 2020). (Any equivalent operator-norm bound yields the same rates up to constants.)

Transformers and attention-based GNNs violate (A1) and are therefore excluded from Theorem 1. By contrast, Theorem 2 requires only adjacency locality and bounded layer operators, and thus extends to adjacency-masked attention under suitable norm bounds (see Remarks 2).

We assume ReLU activations, standard in GCNs, GATs, and GraphSAGE; our minimax bounds remain valid for any larger hypothesis class obtained by replacing ReLU with more expressive or injective MLPs.

We define $\mathcal{F}_{\text{GNN}}(v_s, L)$ as the class of L-layer ReLU GNNs satisfying this constraint. For simplicity, we fix (v_s, L) and write $\mathcal{F}_{\text{GNN}}$.

Risk notions. We quantify generalization error via minimax risks. Here $f^\star \in \mathcal{F}_{\text{GNN}}$ denotes a target function (ground truth), and $\hat{f}$ a learned estimator depending on training data.

Graph-level (inductive) risk: Let $(G_i, X_i, Y_i)_{i=1}^n$ be i.i.d. training samples, where each G_i is an independent graph. Define

$$\mathcal{R}_n^{\text{graph}}(\mathcal{F}_{\text{GNN}}) := \inf_{\hat{f}} \sup_{f^\star \in \mathcal{F}_{\text{GNN}}} \mathbb{E}_{\text{train}} \mathbb{E}_{G \sim \mathbb{P}_G}\left[\left(\hat{f}(G) - f^\star(G)\right)^2\right], \tag{2}$$

where $\mathbb{E}_{\text{train}}$ is over the training graphs $(G_i, X_i, Y_i)_{i=1}^n \sim \mathbb{P}^n$ and the inner expectation is over an independent test graph $G \sim \mathbb{P}_G$.

Node-level (transductive) risk: Fix a connected graph $G = (V, E)$ with features X. Let $S \subset V$ be a uniformly random set of n labeled nodes for training, and let $\hat{f} = \hat{f}(\,\cdot\,; G, X, S)$ denote the learned predictor. Define

$$\mathcal{R}_{(n,G)}^{\text{node}}(\mathcal{F}_{\text{GNN}}) := \inf_{\hat{f}} \sup_{f^\star \in \mathcal{F}_{\text{GNN}}} \mathbb{E}_S\left[\frac{1}{|V|} \sum_{v \in V} \left(\hat{f}(v) - f^\star(v)\right)^2\right], \tag{3}$$

where $\mathbb{E}_S$ is over the random choice of labeled nodes S. Here n counts labeled nodes (not graphs).

These risks correspond to the inductive (graph-level) and transductive (node-level) settings. We will state explicitly which risk each theorem concerns.

Our first theoretical contribution yields a lower bound on the graph-level (inductive) risk.

Theorem 1 (Graph-level Minimax Lower Bound (Inductive)). *Let $\mathcal{F}_{\mathrm{GNN}}$ be the class of L-layer ReLU GNNs with weights satisfying $\sum_{\ell=0}^{L-1}(\|W^{(\ell)}\|_1 + \|B^{(\ell)}\|_1) \le v_s$, with $L \ge 1$ and $v_s > 0$. Assume $(G_i, X_i, Y_i)_{i=1}^n$ are i.i.d. samples with $Y_i = f^\star(G_i, X_i) + U_i$, $U_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$, $f^\star \in \mathcal{F}_{\mathrm{GNN}}$. Then there exists a constant $K_{new} > 0$ such that, for all $n \ge 1$ and $d \ge 2$,*

$$\mathcal{R}_n^{\mathrm{graph}}(\mathcal{F}_{\mathrm{GNN}}) \;\ge\; K_{new}\frac{\sigma v_s}{L}\sqrt{\frac{\log d}{n}}. \tag{4}$$

Interpretation of Theorem 1. The risk decays no faster than $1/\sqrt{n}$, matching classical results for fully connected ReLU networks (Golestaneh et al., 2024).

Sample-size implication. To guarantee error at most ϵ^2, one must have

$$\epsilon^2 \;\ge\; K_{\text{new}}\frac{\sigma v_s}{L}\sqrt{\frac{\log d}{n}} \;\Longrightarrow\; n \;\ge\; K_{\text{new}}^2\frac{\sigma^2 v_s^2}{L^2}\,\frac{\log d}{\epsilon^4}. \tag{5}$$

Compared to classical finite-dimensional parametric estimators (e.g., linear regression, where $n \ge \sigma^2/\epsilon^2$), GNNs require substantially more data to achieve comparable generalization guarantees.

Proof Sketch. We apply Fano's inequality (Fano & Hawkins, 1961) and construct a packing set $\mathcal{M} \subset \mathcal{F}_{\mathrm{GNN}}$ by varying the first-layer weights $W^{(0)}$ on path (chain) graphs. Exhibiting hardness on one such family suffices to establish a minimax lower bound for the unrestricted graph class. Node features are sampled as $X_i \sim \mathcal{N}(0, I_d)$, and labels follow $Y_i = f^*(G_i, X_i) + U_i$, with $U_i \sim \mathcal{N}(0, \sigma^2)$.

Packing step. The bound relies on Lemma 1, which constructs a constant-weight Varshamov–Gilbert code realized by first-layer coordinate selectors and shows

$$\log \mathcal{M}(2\epsilon, \mathcal{F}_{\mathrm{GNN}}, \|\cdot\|_{L_2}) \;\;\ge\;\; \frac{C_A v_s^2 \log d}{L^2\epsilon^2} \tag{6}$$

Applying Fano's inequality with KL divergence bounded by $\mathrm{KL}(P_j\|P_k) \le \frac{2\epsilon^2}{\sigma^2}$ yields

$$\mathcal{R}_{(n,|V|)} \ge \frac{\epsilon^2}{2}\left(1 - \frac{2n\epsilon^2/\sigma^2 + \log 2}{C_A v_s^2 \log d / L^2\epsilon^2}\right). \tag{7}$$

Optimizing over ϵ^2 gives the desired bound. The complete proof is provided in Appendix B.

Remark 1 (Worst-case graphs). *Theorem 1 is established on* path graphs *(chain graphs), where each node has degree at most two. This minimal connectivity creates bottlenecks that slow message passing, making depth the dominant factor. Path graphs thus serve as canonical* worst-case instances*: hardness on this sparse structure certifies the lower bound for all admissible graphs. Although denser graphs may empirically converge faster, the path graph ensures the universal worst-case rate.*

Remark 2 (Exclusion of attention in Theorem 1). *The packing construction for Theorem 1 exploits assumption (A1), i.e., input-independent local aggregation. Architectures with attention violate (A1) because their mixing weights depend on hidden features; hence the theorem does not apply to graph transformers or attention-based GNNs. This does not contradict the lower bound: by monotonicity of minimax risk, enlarging the hypothesis class cannot reduce the bound.*

Theorem 1 establishes $\sqrt{\frac{\log d}{n}}$ scaling, whereas our empirical results (Section 5) indicate $1/\log n$ scaling in practice. This motivates a refined lower bound under structural graph assumptions, formalized in Theorem 2. We first define the notion of *Spectral–homophily* used therein.

Spectral–homophily. The induced labeled-node subgraph satisfies $\lambda_2(\mathcal{L}_n) \le \kappa/\log n$, a *structural* expansion/mixing condition (small spectral gap), distinct from label-homophily assumptions (see Appendix C).

Theorem 2 (Structured-Graph Minimax Lower Bound (Node-Level, Transductive)). *Let $L \ge 1$, $v_s > 0$, and let $G = (V, E)$ satisfy the* spectral–homophily *condition $\lambda_2(\mathcal{L}) \le \kappa/\log n$ for some universal $\kappa > 0$, where n is the number of labeled training nodes and $\mathcal{L}$ is the normalized Laplacian. Then there exists a universal constant $\Gamma > 0$ such that*

$$\mathcal{R}_{(n,G)}^{\mathrm{node}}(\mathcal{F}_{\mathrm{GNN}}) \;\ge\; \frac{\sigma^2 v_s^2}{\Gamma L^2}\cdot\frac{d}{\log n}. \tag{8}$$

Interpretation of Theorem 2. This bound decays more slowly than $1/\sqrt{n}$, making it tighter whenever the spectral–homophily condition holds (see Eq. (22) and Appendix H for an explicit form). Extensions to adjacency-masked attention (e.g., GAT) are discussed in Appendices I–J, and practical guidance on improving constants without changing the $\Omega(d/\log n)$ rate is in Appendix K. If spectral–homophily condition fails (e.g., λ_2 is larger, indicating strong expansion), the independence argument breaks down and the analysis reverts to Theorem 1, yielding the $\Omega(\sqrt{\log d/n})$ rate.

Sample-size implication. To achieve generalization error ϵ^2, the following must hold:

$$\frac{\sigma^2 v_s^2}{\Gamma L^2}\,\frac{d}{\log n} \;\le\; \epsilon^2 \quad\Longrightarrow\quad n \;\ge\; \exp\!\Big(\tfrac{\sigma^2 v_s^2 d}{\Gamma L^2 \epsilon^2}\Big), \tag{9}$$

implying exponential sample complexity in $1/\epsilon^2$, far worse than polynomial rates.

4 Structured–Graph Lower Bound (Proof of Theorem 2)

Proof. Consider the node-level transductive setting of Eq. (3) on a fixed graph $G = (V, E)$, with each training example corresponding to a distinct node. We impose the following **spectral–homophily condition** on the subgraph induced by the n training nodes: $\lambda_2\big(\mathcal{L}_n\big) \;\le\; \frac{\kappa}{\log n}$, where $\mathcal{L}_n$ is the normalized Laplacian and $\kappa > 0$ is universal. By Lemma 3 (Appendix F), the induced subgraph has random-walk mixing time $O(\log n)$. Consequently, message-passing neighborhoods overlap heavily, and only $\Theta(\log n)$ samples provide nearly independent signal. Intuitively, after $O(\log n)$ steps the graph "looks new," so only one out of every $\Theta(\log n)$ samples contributes fresh information. The proof formalizes this intuition in four steps.

Block decomposition. Fix $\varepsilon \in (0, 1)$, say $\varepsilon = \frac{1}{4}$. By Lemma 3, if $\lambda_2 \le \kappa/\log n$, then the random walk on the induced subgraph mixes in time $t_{\mathrm{mix}}(\varepsilon) = O(\log n)$, with constants depending only on κ, ε, and the laziness parameter. Let $K = K(\lambda_2, \kappa)$ denote the effective number of nearly independent blocks obtained from the mixing-time argument. In particular, under $\lambda_2 \le \kappa/\log n$, we have $K = \Theta(\log n)$; for concreteness we write $K := \lceil C_{\mathrm{mix}} \log n \rceil$ for a suitable constant $C_{\mathrm{mix}} > 0$. Then $K = \Theta(\log n)$. Select K nodes $\{i_1, \ldots, i_K\}$ separated by at least the mixing radius (graph distance $\gtrsim \log n$). The corresponding outputs $Y_{i_1}, \ldots, Y_{i_K}$ are then approximately independent when evaluated on appropriately localized functions $f^\star$. A typical consequence is that covariances decay rapidly with separation, e.g. $\big|\mathrm{Cov}(Y_{i_\ell}, Y_{i_{\ell'}})\big| \le \sigma^2 e^{-c\,\mathrm{dist}(i_\ell, i_{\ell'})}$ where $\mathrm{dist}(i_\ell, i_{\ell'})$ is large. We tie block ℓ exclusively to node i_ℓ; for the constructed functions $f_{\mathbf{s}}$, support is restricted to these K nodes.

Step 1: Sparse packing across blocks. Define $h(\varepsilon) := \lceil 16 L^2 K^2 \varepsilon^2 / v_s^2 \rceil$. Construct a codebook $\mathcal{C} \subset \{0, 1\}^d$ of weight-$d/4$ vectors with pairwise Hamming distance at least $h(\varepsilon)$. Existence is guaranteed by the Gilbert–Varshamov bound (Varshamov, 1957; Gilbert, 1952). Assign each block $\ell = 1, \ldots, K$ a codeword $s^{(\ell)} \in \mathcal{C}$ and let $\boldsymbol{s} = (s^{(1)}, \ldots, s^{(K)})$. Define

$$f_{\boldsymbol{s}}(x) := \sum_{\ell=1}^{K} \frac{v_s}{LK} \sum_{j=1}^{d} s_j^{(\ell)}\, \phi(x_j), \qquad \phi(z) = \max\{0, z\}. \tag{10}$$

This function is realized by a one-layer ReLU GNN with self-loops and an identity aggregator (so each node aggregates only its own features). Hence $f_{\mathbf{s}} \in \mathcal{F}_{\mathrm{GNN}}(v_s, 1)$; see Appendix A. Its complexity, determined by the magnitude of its coefficients (e.g., $\sum_{\ell,j} \big|\frac{v_s}{LK} s_j^{(\ell)}\big| = \frac{v_s d}{4L}$), is therefore bounded consistently with the definition of $\mathcal{F}_{\mathrm{GNN}}$, as parameterized by v_s and L.

Separation. Suppose $\boldsymbol{s}$ and $\boldsymbol{s}'$ differ only in block m. Then $f_{\boldsymbol{s}}(x) - f_{\boldsymbol{s}'}(x) = \frac{v_s}{LK} \sum_{j=1}^{d} (s_j^{(m)} - s_j'^{(m)})\, \phi(x_j)$. Hence

$$\|f_{\boldsymbol{s}} - f_{\boldsymbol{s}'}\|_{L_2}^2 = \mathbb{E}_X \left[\left(\tfrac{v_s}{LK} \sum_j (s_j^{(m)} - s_j'^{(m)}) \phi(X_j) \right)^2 \right].$$

Assuming the features $\{\phi(X_j)\}$ are orthonormal, this simplifies to

$$\|f_{\boldsymbol{s}} - f_{\boldsymbol{s}'}\|_{L_2}^2 = \tfrac{v_s^2}{L^2 K^2}\, \|s^{(m)} - s'^{(m)}\|_2^2. \tag{11}$$

By the codebook construction, $\|s^{(m)} - s'^{(m)}\|_2^2 \geq h(\varepsilon)$. Substituting into Eq. (11) and recalling $h(\varepsilon) \geq \frac{16L^2K^2\varepsilon^2}{v_s^2}$ yields $\|f_{\boldsymbol{s}} - f_{\boldsymbol{s}'}\|_{L_2}^2 \geq 16\varepsilon^2$. Thus any two functions differing in one block are separated by at least $16\varepsilon^2$.

Packing Set Construction for Fano's Inequality. To apply Fano's inequality (Lemma 2, Appendix D), we construct a set of M functions $\{f_{\mathbf{x}}\}$ from $\mathcal{F}_{\text{GNN}}$ that are well separated in L_2 norm yet induce output distributions that are not too distinguishable.

Let $\Gamma_c > 0$ be a sufficiently large universal constant (its value will be fixed by the conditions below and will enter the final constant Γ in the theorem). Define a target Hamming distance for length-d codewords:

$$\Delta_H := \left\lceil \frac{16\sigma^2 d}{\Gamma_c} \right\rceil. \tag{12}$$

We require Γ_c large enough (e.g., $\Gamma_c > 64\sigma^2$, for $d \geq 1$) so that $\Delta_H \leq d/4$. This guarantees the existence of two codewords $s_0, s_1 \in \{0,1\}^d$ such that: (i) $\|s_0\|_0 = \|s_1\|_0 = d/4$ (both have weight $d/4$), and (ii) $\|s_0 - s_1\|_2^2 = d_H(s_0, s_1) \geq \Delta_H$. The existence of such constant-weight codewords follows from standard coding theory results.

Now let $K = \lceil \log n \rceil$. For $K \geq 4$, the Varshamov–Gilbert bound ensures the existence of a code $\mathcal{C}_K \subset \{0,1\}^K$ of size $M = |\mathcal{C}_K|$ with pairwise Hamming distance at least $K/4$, i.e., $d_H(\mathbf{x}, \mathbf{x}') \geq K/4$, and $\log M \geq c_1 K$ for some universal $c_1 > 0$.

For each $\mathbf{x} = (x_1, \ldots, x_K) \in \mathcal{C}_K$, define a function $f_{\mathbf{x}} \in \mathcal{F}_{\text{GNN}}$ as follows. For each of the K special nodes $\{i_1, \ldots, i_K\}$, assign block ℓ (tied to node i_ℓ) the codeword

$$s_{\mathbf{x}}^{(\ell)} = \begin{cases} s_1, & \text{if } x_\ell = 1, \\ s_0, & \text{if } x_\ell = 0, \end{cases} \tag{13}$$

and set $f_{\mathbf{x}}(X_{i_\ell}) = \frac{v_s}{LK} \sum_{j=1}^d (s_{\mathbf{x}}^{(\ell)})_j \, \phi((X_{i_\ell})_j)$ and $f_{\mathbf{x}}(X_p) = 0$ for $p \notin \{i_1, \ldots, i_K\}$.

The squared L_2-distance between two such functions $f_{\mathbf{x}}$ and $f_{\mathbf{x}'}$ is

$$\|f_{\mathbf{x}} - f_{\mathbf{x}'}\|_{L_2}^2 = \sum_{p=1}^{|V|} \big(f_{\mathbf{x}}(X_p) - f_{\mathbf{x}'}(X_p)\big)^2 = \sum_{\ell=1}^K \left(\frac{v_s}{LK} \sum_{j=1}^d \big((s_{\mathbf{x}}^{(\ell)})_j - (s_{\mathbf{x}'}^{(\ell)})_j\big) \phi((X_{i_\ell})_j) \right)^2. \tag{14}$$

Assuming orthonormal features $\{\phi((X_{i_\ell})_j)\}$ (as in the separation argument), this simplifies to

$$\begin{aligned} \|f_{\mathbf{x}} - f_{\mathbf{x}'}\|_{L_2}^2 &= \sum_{\ell=1}^K \left(\frac{v_s}{LK}\right)^2 \|s_{\mathbf{x}}^{(\ell)} - s_{\mathbf{x}'}^{(\ell)}\|_2^2 \\ &= d_H(\mathbf{x}, \mathbf{x}') \left(\frac{v_s}{LK}\right)^2 \|s_1 - s_0\|_2^2 \geq \frac{K}{4} \left(\frac{v_s}{LK}\right)^2 \Delta_H = \frac{\Delta_H v_s^2}{4L^2K}. \end{aligned} \tag{15}$$

Thus the minimum squared separation is $d_0^2 = \frac{\Delta_H v_s^2}{4L^2K}$.

Step 2: KL divergence. Let $P_{\mathbf{x}}$ be the distribution of the observations $Y = (Y_{i_1}, \ldots, Y_{i_K})$ when the true function is $f_{\mathbf{x}}$ and each Y_{i_ℓ} is corrupted by independent Gaussian noise $N(0, \sigma^2)$. The KL divergence between $P_{\mathbf{x}}$ and $P_{\mathbf{x}'}$ is

$$\begin{aligned} KL(P_{\mathbf{x}} \| P_{\mathbf{x}'}) &= \sum_{\ell=1}^K \frac{1}{2\sigma^2} \left(f_{\mathbf{x}}(X_{i_\ell}) - f_{\mathbf{x}'}(X_{i_\ell})\right)^2 = \frac{1}{2\sigma^2} \|f_{\mathbf{x}} - f_{\mathbf{x}'}\|_{L_2(\text{on } K \text{ nodes})}^2 \\ &= \frac{1}{2\sigma^2} d_H(\mathbf{x}, \mathbf{x}') \left(\frac{v_s}{LK}\right)^2 \|s_1 - s_0\|_2^2 \leq \frac{K}{2\sigma^2} \left(\frac{v_s}{LK}\right)^2 \Delta_H = \frac{\Delta_H v_s^2}{2\sigma^2 L^2 K}. \end{aligned} \tag{16}$$

Thus $KL_{\max} := \frac{\Delta_H v_s^2}{2\sigma^2 L^2 K}$.

Step 3: Fano's inequality. Applying Lemma 2 (Fano–Tsybakov; see Appendix D), if we have M functions $\{f_{\mathbf{x}}\}_{\mathbf{x} \in \mathcal{C}_K}$ such that $\|f_{\mathbf{x}} - f_{\mathbf{x}'}\|_{L_2}^2 \geq d_0^2$ for all $\mathbf{x} \neq \mathbf{x}'$ and $KL(P_{\mathbf{x}} \| P_{\mathbf{x}'}) \leq KL_{\max}$, then

$$\inf_{\hat{f}} \sup_{\mathbf{x} \in \mathcal{C}_K} \mathbb{E}[\|\hat{f} - f_{\mathbf{x}}\|_{L_2}^2] \geq \frac{d_0^2}{2} \left(1 - \frac{KL_{\max} + \log 2}{\log M}\right). \tag{17}$$

(Some versions yield $d_0^2/8$ under the stronger assumption $KL_{\max} \leq \frac{\log M}{2} - \log 2$; we state the general form.)

To ensure the parenthesis is bounded below by a positive constant, say $c_2 = 1/2$, we require $\log M \geq 2(KL_{\max} + \log 2)$. Since $\log M \geq c_1 K$, this condition reduces to

$$c_1 K \geq \frac{\Delta_H v_s^2}{\sigma^2 L^2 K} + 2\log 2. \tag{18}$$

When Eq. (18) holds, the minimax risk satisfies $\mathcal{R}^{\text{node}}_{(n,G)}(\mathcal{F}_{\text{GNN}}) \geq \frac{c_2 d_0^2}{2} = \frac{c_2}{2} \cdot \frac{\Delta_H v_s^2}{4L^2 K}$. Substituting $\Delta_H = \lceil 16\sigma^2 d/\Gamma_c \rceil \geq 16\sigma^2 d/\Gamma_c$ gives

$$\mathcal{R}^{\text{node}}_{(n,G)}(\mathcal{F}_{\text{GNN}}) \geq \frac{c_2}{2} \cdot \frac{(16\sigma^2 d/\Gamma_c)\, v_s^2}{4L^2 K} = \frac{2c_2}{\Gamma_c} \cdot \frac{\sigma^2 v_s^2 d}{L^2 \log n} = \frac{\sigma^2 v_s^2}{\Gamma L^2} \cdot \frac{d}{\log n}, \tag{19}$$

where $\Gamma := \Gamma_c/(2c_2)$. This completes the proof.

Appendix G derives sufficient conditions on Γ_c to ensure Eq. (18) holds, confirming that $\Gamma = \Gamma_c/(2c_2)$ is a universal constant. These calculations refine the constants and verify the claimed scaling. □

5 Empirical Studies

In this section, we provide proof-of-concept experiments to assess how our theoretical results align with practice. Experiments were conducted on four benchmark datasets—Cora, Reddit, QM9, and Facebook—using three representative GNN architectures: GCN (Kipf & Welling, 2017), GAT (Veličković et al., 2018), and GraphSAGE (Hamilton et al., 2017). Dataset descriptions, training protocols, and infrastructure details appear in Appendix M.

Scope. Our theory establishes bounds for graph-level (Theorem 1) and node-level (Theorem 2) prediction. For completeness we also report one link-level task (Facebook), though no formal bound is provided. Theorem 1 applies to local, input-independent aggregation, while Theorem 2 extends to adjacency-masked attention (standard GAT) under bounded layer norms—conditions satisfied by our GAT implementation.

Across 7 of 12 dataset–model combinations, the observed minimax risk (generalization error) decays closer to $1/\log(n)$ than the $1/\sqrt{n}$ rate of Theorem 1. This pattern suggests that the refined bound in Theorem 2 may better capture empirical GNN behavior in several settings.

Graph Structural Properties. Table 1 reports structural statistics— homophily (fraction of same-label edges) and spectral gap (second Laplacian eigenvalue; see Appendix M.1)— and their relation to observed scaling. Datasets with higher homophily, such as Cora (0.81) and Reddit (0.78), tend to show $1/\log(n)$ scaling, consistent with node similarity aiding information propagation. In contrast, Facebook, with lower homophily (0.58) and a smaller spectral gap ($\lambda_2 = 0.05$), more often follows $1/\sqrt{n}$ scaling in link prediction, suggesting weaker diffusion and limited regularization.

Two exceptions—GCN on Reddit and GraphSAGE on QM9—occur in settings may hinder global regularization: very large graphs ($|V| > 10^5$) or molecular tasks needing fine-grained distinctions. The observed dependence on homophily and spectral gaps aligns with the spectral-homophily condition in Theorem 2, which predicts slower convergence when structural regularization is weak.

Table 1: Graph Structural Properties Supporting Theorem 2

Dataset	**Homophily Ratio**	**Clustering Coefficient**	**Spectral Gap** (λ_2)
Cora	0.81	0.24	0.12
Reddit	0.78	0.15	0.08
QM9	N/A	0.03	0.18
Facebook	0.58	0.30	0.05

Methodology. We implemented all models in PyTorch Geometric and trained them on sample sizes $n \in \{100, 500, 1k, 5k, 10k, 50k\}$, subject to dataset limits ($n \leq 1{,}000$ for Cora and Facebook). For

each n, we computed test errors averaged over five runs with different random seeds. Classification tasks (Cora, Reddit) used cross-entropy loss, regression (QM9) used mean squared error, and link prediction (Facebook) used one minus AUC (1-AUC). For link prediction, edges were randomly sampled into balanced positive/negative sets of size n. To analyze error scaling, we fitted four candidate forms to the test error curves: $c_1 + \frac{\alpha}{\sqrt{n}}$, $c_2 + \frac{\beta}{n}$, $c_3 + \frac{\delta}{\log n}$, $c_4 + \frac{1}{n^\gamma}$. Parameters $(c_1, c_2, c_3, c_4, \alpha, \beta, \delta, \gamma \in (0, \infty))$ were optimized via weighted least-squares regression with inverse-variance weights. Fit quality was evaluated using residual sum of squares (RSS), mean squared error (MSE), and the coefficient of determination (R^2), to capture both absolute and relative fit quality.

Results. Table 2 summarizes fit quality across all datasets and models, and Figures 1–4 compare fitted curves with empirical errors. The best-fit scaling law varies by dataset and architecture. Broadly, $1/\log(n)$ dominates in high-homophily settings (e.g., Cora, several cases in Reddit and QM9), while $1/\sqrt{n}$ better explains performance in some large-scale or regression tasks (e.g., GCN on Reddit, GAT on QM9). The flexible $1/n^\gamma$ form occasionally yields the best fit, most notably for GAT on Facebook with $\gamma \approx 0.42$. In some cases, however, the $1/n^\gamma$ form produces negative R^2 values (Table 2), which simply indicate fits worse than the mean baseline and are not interpreted further. In contrast, the $1/n$ law consistently underperforms.

Table 2: Comparison of Fit Metrics Across All Models and Datasets (Weighted Analysis)

Dataset	Model	$c_1 + \frac{\alpha}{\sqrt{n}}$			$c_2 + \frac{\beta}{n}$			$c_3 + \frac{\delta}{\log n}$			$c_4 + \frac{1}{n^\gamma}$			γ	Best Fit
		RSS	MSE	R^2	RSS	MSE	R^2	RSS	MSE	R^2	RSS	MSE	R^2		
Cora	GCN	1.13e+00	3.76e-01	0.991	6.28e+00	2.09e+00	0.952	2.53e-01	8.45e-02	0.998	3.43e+01	1.14e+01	0.736	0.243	1/log(n)
Cora	GAT	6.71e-02	2.24e-02	0.999	1.41e+00	4.71e-01	0.972	9.12e-03	3.04e-03	1.000	9.87e+00	3.29e+00	0.804	0.245	1/log(n)
Cora	GraphSAGE	1.52e+00	5.08e-01	0.990	9.36e+00	3.12e+00	0.938	2.62e-01	8.73e-02	0.998	5.30e+01	1.77e+01	0.649	0.220	1/log(n)
Reddit	GCN	5.65e+01	9.42e+00	0.835	7.52e+01	1.25e+01	0.781	1.44e+02	2.40e+01	0.580	1.30e+02	2.17e+01	0.621	0.304	1/sqrt(n)
Reddit	GAT	7.13e+01	1.19e+01	0.762	1.87e+02	3.12e+01	0.374	2.89e+01	4.81e+00	0.904	1.16e+02	1.93e+01	0.612	0.120	1/log(n)
Reddit	GraphSAGE	2.41e+02	4.02e+01	0.983	8.25e+03	1.38e+03	0.427	9.58e+01	1.60e+01	0.993	4.44e+03	7.39e+02	0.692	0.227	1/log(n)
QM9	GCN	4.94e+00	8.23e-01	0.928	2.49e+01	4.15e+00	0.635	2.93e+00	4.88e-01	0.957	1.62e+02	2.70e+01	-1.377	0.100	1/log(n)
QM9	GAT	8.78e-01	1.46e-01	0.790	2.12e+00	3.53e-01	0.493	1.10e+00	1.84e-01	0.736	9.52e-01	1.59e-01	0.772	0.360	1/sqrt(n)
QM9	GraphSAGE	1.69e-01	2.81e-02	0.990	3.05e+00	5.09e-01	0.816	1.83e+00	3.05e-01	0.890	2.29e+01	3.81e+00	-0.378	0.100	1/sqrt(n)
Facebook	GCN	4.61e-01	1.54e-01	0.978	1.56e+00	5.19e-01	0.927	1.91e-01	6.37e-02	0.991	5.70e-01	1.90e-01	0.973	0.552	1/log(n)
Facebook	GAT	2.12e-02	7.06e-03	0.998	5.60e-01	1.87e-01	0.954	7.11e-03	2.37e-03	0.999	8.95e-04	2.98e-04	1.000	0.420	$1/n^\gamma$
Facebook	GraphSAGE	1.77e-02	5.92e-03	0.999	2.45e-01	8.16e-02	0.991	1.39e-01	4.64e-02	0.995	7.72e-01	2.57e-01	0.971	0.449	1/sqrt(n)

Different architectures often show different slopes on the same dataset, a phenomenon likely influenced by smoothing, overlap, and bias–variance tradeoffs (Appendix L).

Overall, the empirical results show that convergence rates frequently decay more slowly than the classical $1/\sqrt{n}$ bound, often approaching $1/\log(n)$. This slower rate is consistent with Theorem 2, which predicts weaker convergence under limited structural regularization. In practice, this implies that GNNs may require substantially more data to generalize effectively on graphs with small spectral gaps or weak homophily.

The clear emergence of $1/\log(n)$ scaling on Cora across all models (Figure 1) exemplifies this trend: community structure and weak spectral gap appear to constrain diffusion, raising sample complexity beyond the optimistic $1/\sqrt{n}$ rate. On Reddit, GCN fits $1/\sqrt{n}$ while GAT and GraphSAGE align with $1/\log(n)$, suggesting that architectural sophistication alone may not offset structural limits. On QM9, regression tasks show mixed scaling, while Facebook link prediction exhibits task-specific behavior, with GAT following $1/n^\gamma$.

Taken together, these results highlight the importance of structure-aware generalization bounds that account for both graph topology and task characteristics, rather than relying solely on universal rates.

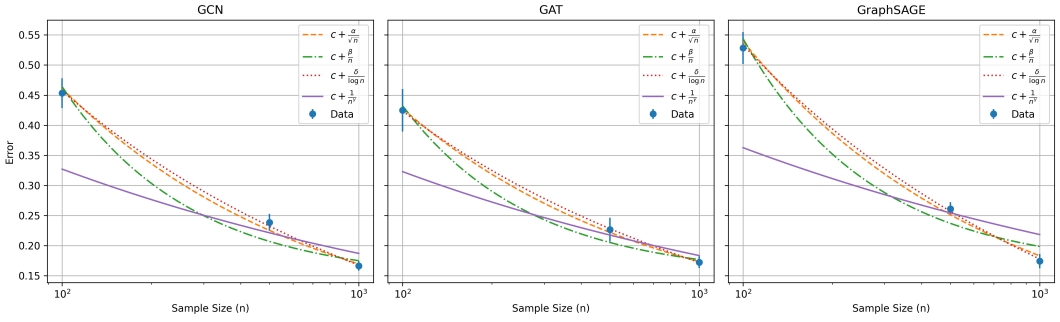

Figure 1: Test error vs. sample size n on Cora.

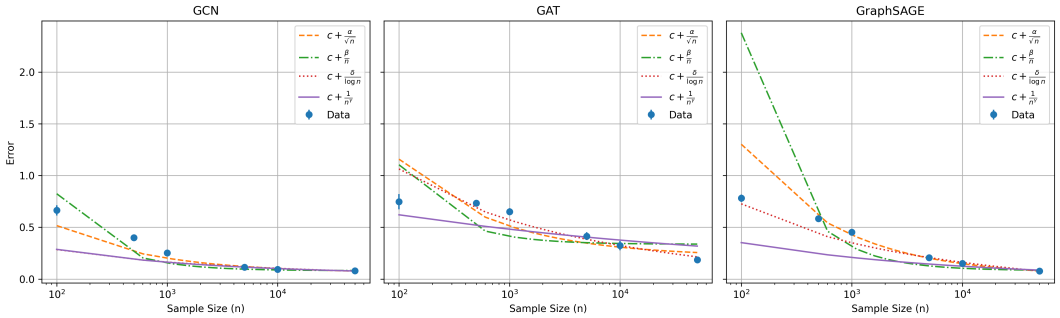

Figure 2: Test error vs. sample size n on Reddit.

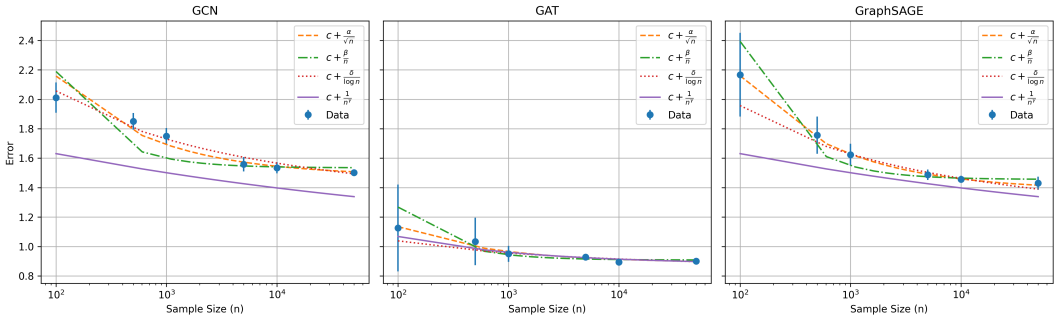

Figure 3: Test error vs. sample size n on QM9.

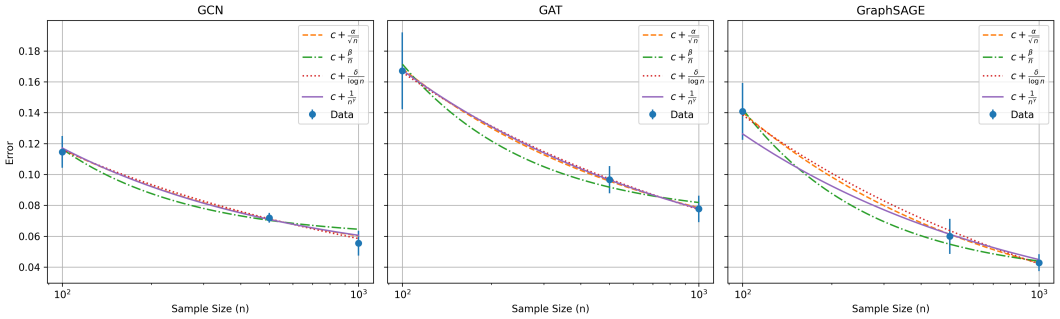

Figure 4: Test error vs. sample size n on Facebook.

6 Conclusion

We develop a theoretical foundation for the sample complexity of ReLU-based Graph Neural Networks (GNNs), addressing a central gap in understanding their statistical limits. Using minimax analysis, we show that while GNNs can in principle match the $1/\sqrt{n}$ scaling of feed-forward networks, realistic structural assumptions—such as strong homophily and bounded spectral expansion—force risk to decay no faster than $1/\log(n)$. This implies that reliable generalization on structured data may require substantially more samples than previously assumed.

Empirical studies on four benchmarks and three architectures support this refined picture: in most regimes with community structure or small spectral gaps, generalization follows the slower $1/\log(n)$ rate rather than the classical $1/\sqrt{n}$. These results identify graph topology as a primary driver of sample efficiency, beyond architectural design alone.

In sum, we provide the first sharp lower bounds for GNNs under realistic structures, together with empirical evidence that these slower rates arise in practice. Future work should investigate whether alternative architectures, regularization, or pre-training can overcome the inherent data inefficiency induced by weak homophily and limited spectral expansion.

7 Reproducibility Statement

We have taken several steps to ensure the reproducibility of our work. All theoretical assumptions, theorems, and the proof sketch of Theorem 1 are explicitly stated in Section 3. The complete proofs of Theorem 1 and Theorem 2 are provided in Appendix B and Section 4, respectively. Supporting technical components—including degenerate GNN realizations (Appendix A), spectral and homophily assumptions (Appendix C), Fano's inequality (Appendix D), mixing-time arguments (Appendix F), and operator-norm control for attention (Appendices I–J)—are all provided for completeness. Experimental protocols are described in Section 5, while dataset descriptions, training procedures, and infrastructure details appear in Appendix M. To further support verification, we provide the full source code as supplementary material, including implementations for data loading, model training, evaluation, and error analysis. The package also contains scripts to reproduce all experimental results, regenerate LaTeX tables, and visualize learning curves. Together, these resources ensure that both the theoretical and empirical results reported in this paper can be independently reproduced and validated.

References

Afonso S Bandeira, Amit Singer, and Daniel A Spielman. A cheeger inequality for the graph connection laplacian. *SIAM Journal on Matrix Analysis and Applications*, 34(4):1611–1630, 2013.

Robert M Fano and David Hawkins. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794, 1961.

Billy Joe Franks, Christopher Morris, Ameya Velingker, and Floris Geerts. Weisfeiler-leman at the margin: When more expressivity matters. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 13885–13926. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/franks24a.html.

Vikas K. Garg, Stefanie Jegelka, and Tommi Jaakkola. Generalization and representational limits of graph neural networks, 2020. URL https://arxiv.org/abs/2002.06157.

Edgar N Gilbert. A comparison of signalling alphabets. *The Bell system technical journal*, 31(3): 504–522, 1952.

Pegah Golestaneh, Mahsa Taheri, and Johannes Lederer. How many samples are needed to train a deep neural network?, 2024. URL https://arxiv.org/abs/2405.16696.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/5dd9db5e033da9c6fb5ba83c7a7ebea9-Paper.pdf.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2017. URL https://arxiv.org/abs/1609.02907.

Johannes Lederer. Statistical guarantees for sparse deep learning, 2022. URL https://arxiv.org/abs/2212.05427.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

Renjie Liao, Raquel Urtasun, and Richard Zemel. A pac-bayesian approach to generalization bounds for graph neural networks, 2020. URL https://arxiv.org/abs/2012.07690.

Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

Giannis Nikolentzos, George Dasoulas, and Michalis Vazirgiannis. Permute me softly: Learning soft permutations for graph representations, 2022. URL https://arxiv.org/abs/2110.01872.

Kenta Oono and Taiji Suzuki. Graph neural networks exponentially lose expressive power for node classification, 2021. URL https://arxiv.org/abs/1905.10947.

Paolo Pellizzoni, Till Hendrik Schulz, Dexiong Chen, and Karsten Borgwardt. On the expressivity and sample complexity of node-individualized graph neural networks. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014a.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1(1):1–7, 2014b. doi: 10.1038/sdata.2014.22.

Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2289–2292, 2019. doi: 10.1145/3357384.3357939.

Lars Ruddigkeit, Ruud Van Deursen, Lorenz C Blum, and Jean-Louis Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling*, 52(11):2864–2875, 2012.

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, 2009. doi: 10.1109/TNN.2008.2005605.

Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with relu activation function. *The Annals of Statistics*, 48(4), August 2020. ISSN 0090-5364. doi: 10.1214/19-aos1875. URL http://dx.doi.org/10.1214/19-AOS1875.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93, Sep. 2008. doi: 10.1609/aimag.v29i3.2157. URL https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2157.

Mahsa Taheri, Fang Xie, and Johannes Lederer. Statistical guarantees for regularized neural networks, 2020. URL https://arxiv.org/abs/2006.00294.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Series in Statistics. Springer, 2009. See Lemma 2.10, Chapter 2.

Rom Rubenovich Varshamov. Estimate of the number of signals in error correcting codes. *Docklady Akad. Nauk, SSSR*, 117:739–741, 1957.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018. URL https://arxiv.org/abs/1710.10903.

A Degenerate GNN Realization

We construct a one-layer ReLU GNN on the original graph (with self-loops) using the identity aggregator, $\mathrm{Agg} = \mathrm{identity}$. In this case, each node aggregates only its own features—a degenerate but still admissible instance of the message passing. With weights set as $W_j = \frac{v_s}{LK} s_j^{(\ell)}$ and zero bias, the network output is

$$f_{\boldsymbol{s}}(x) = \sum_{\ell=1}^{K} \frac{v_s}{LK} \sum_{j=1}^{d} s_j^{(\ell)}\, \phi(x_j),$$

which lies in $\mathcal{F}_{\mathrm{GNN}}(v_s, 1)$. Although message passing here reduces to self-loops, this subclass is included in our hypothesis space. Since minimax lower bounds apply to any subclass, establishing hardness for these degenerate cases certifies hardness for the full class.

B MINIMAX LOWER BOUND (PROOF OF THEOREM 1)

We begin with a technical packing lemma, which establishes the key combinatorial bound used in Step 1 of the proof of Theorem 1.

Lemma 1 (Packing for ReLU under Gaussian features)**.** *Let $X \sim \mathcal{N}(0, I_d)$ and $\phi(z) = \max\{0, z\}$. Consider $\mathcal{F}_{\mathrm{GNN}}(v_s, L)$, the class of L-layer ReLU GNNs with*

$$\sum_{\ell=0}^{L-1} \big(\|W^{(\ell)}\|_1 + \|B^{(\ell)}\|_1\big) \le v_s.$$

There exist absolute constants $c, C_A > 0$ such that for every $\epsilon \in \big(0,\, c\, v_s/L\big]$, the 2ϵ-packing number of $\mathcal{F}_{\mathrm{GNN}}(v_s, L)$ with respect to the $L_2(P_X)$ metric satisfies

$$\log \mathcal{M}\big(2\epsilon,\, \mathcal{F}_{\mathrm{GNN}}(v_s, L),\, \|\cdot\|_{L_2(P_X)}\big) \;\ge\; C_A\, \frac{v_s^2}{L^2\, \epsilon^2}\, \log d.$$

Proof. Fix $L \ge 1$ and $v_s > 0$. We construct a family $\{f_S\}$ indexed by r-subsets $S \subset [d]$, for a choice of r defined below, and we show it is a 2ϵ-packing.

(L1) Realizable subclass and budget. Let $r \in \{1, \dots, d\}$ and define

$$f_S(x) \;:=\; a \sum_{j \in S} \phi(x_j) \qquad \text{with} \qquad a \;=\; \frac{c_0\, v_s}{L\, r},$$

where $c_0 \in (0, 1)$ is an absolute constant to be fixed. We claim $f_S \in \mathcal{F}_{\mathrm{GNN}}(v_s, L)$. Realize f_S by using the first layer to compute the r hidden coordinates $\{\phi(x_j) : j \in S\}$ with weights whose ℓ_1 sum is ra (so this layer spends $ra = c_0 v_s/L$ of budget). Use the last layer as a linear readout that sums these r hidden coordinates with weights of total ℓ_1 norm at most v_s/L, and set all intermediate layers to the zero operator. The overall output equals $a \sum_{j \in S} \phi(x_j)$. The total ℓ_1 budget used is at most $(c_0 + 1)\, v_s/L \le v_s$ for $c_0 \le 1$, so $f_S \in \mathcal{F}_{\mathrm{GNN}}(v_s, L)$. (Absolute constants can be absorbed into c_0; no rate is affected.)

(L2) L_2 separation. Let $Z, Z' \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$. Standard ReLU–Gaussian moments give $\mathbb{E}[\phi(Z)] = 1/\sqrt{2\pi}$, $\mathbb{E}[\phi(Z)^2] = 1/2$, and for independent Z, Z', $\mathbb{E}[\phi(Z)\phi(Z')] = 1/(2\pi)$. Hence for $j \ne k$,

$$\mathbb{E}\big[(\phi(X_j) - \phi(X_k))^2\big] = \Big(\tfrac{1}{2} - \tfrac{1}{2\pi}\Big) + \Big(\tfrac{1}{2} - \tfrac{1}{2\pi}\Big) \;\ge\; 1 - \tfrac{1}{\pi} \;=:\; c_\star \in (0, 1).$$

Let $S, T \subset [d]$ with $|S| = |T| = r$, and write $D = S \triangle T$ (symmetric difference), $m := |D|$. By independence across coordinates and the display above,

$$\|f_S - f_T\|_{L_2(P_X)}^2 = a^2\, \mathbb{E}\Big[\Big(\sum_{j \in S} \phi(X_j) - \sum_{k \in T} \phi(X_k)\Big)^2\Big] \;\ge\; \frac{a^2\, m}{2}.$$

(Since cross-covariances between distinct coordinates vanish, we retain only the diagonal terms as a conservative lower bound. Accounting for the exact covariance yields the slightly larger constant $c_\star$ in place of $1/2$, but the simpler factor $1/2$ already provides a valid bound.)

(L3) Constant-weight code. By the Varshamov–Gilbert bound for constant-weight codes, there exists $\mathcal{C} \subset \{S \subset [d] : |S| = r\}$ such that for all distinct $S, T \in \mathcal{C}$, $|S \triangle T| \ge r/2$ and $|\mathcal{C}| \ge (c\, d/r)^r$ for a universal $c \in (0, 1)$. Combining with (L2) gives, for $S \ne T \in \mathcal{C}$,

$$\|f_S - f_T\|_{L_2(P_X)} \;\ge\; \frac{a\sqrt{r}}{2}.$$

(L4) Choosing r to achieve 2ϵ separation. We want $\|f_S - f_T\|_{L_2(P_X)} \ge 2\epsilon$ for all distinct $S, T \in \mathcal{C}$, i.e., $\frac{a\sqrt{r}}{2} \ge 2\epsilon$. With $a = (c_0 v_s)/(Lr)$ this becomes

$$\frac{c_0 v_s}{2L\sqrt{r}} \;\ge\; 2\epsilon \quad\Longleftrightarrow\quad r \;\le\; \frac{c_0^2}{16}\, \frac{v_s^2}{L^2 \epsilon^2}.$$

We take

$$r \;=\; \Big\lfloor \frac{c_0^2}{32}\,\frac{v_s^2}{L^2\epsilon^2} \Big\rfloor \qquad \text{and assume } \epsilon \le c_1 \frac{v_s}{L},$$

with $c_1 > 0$ small enough so that $1 \le r \le d/2$ (thus $\log(d/r) \ge \frac{1}{2}\log d$). Then $\{f_S : S \in \mathcal{C}\}$ is a 2ϵ-packing.

(L5) Packing size. From (L3) and $r \le d/2$ we get

$$\log \mathcal{M}(2\epsilon, \mathcal{F}_{\mathrm{GNN}}, \|\cdot\|_{L_2(P_X)}) \;\ge\; \log|\mathcal{C}| \;\ge\; c' r \log(d/r) \;\ge\; \frac{c'}{2}\, r\, \log d.$$

Substituting the choice of r from (L4) and absorbing absolute constants (including $c_0, c', \frac{1}{2}$, and the ReLU–Gaussian factor) yields

$$\log \mathcal{M}(2\epsilon, \mathcal{F}_{\mathrm{GNN}}, \|\cdot\|_{L_2(P_X)}) \;\ge\; C_A\, \frac{v_s^2}{L^2\epsilon^2}\, \log d,$$

for a universal $C_A > 0$, proving the claim. □

With Lemma 1 established, we now prove Theorem 1.

Proof. The proof proceeds by Fano's inequality, which requires (i) a large packing set inside $\mathcal{F}_{\mathrm{GNN}}$, and (ii) a KL-divergence bound (Appendix D). Step 1 invokes Lemma 1, whose proof appears above.

Step 1: Packing number. By Lemma 1, for every $\epsilon \le c\, v_s/L$, there exists a 2ϵ-packing $\mathcal{M}^* = \{f_1, \dots, f_M\}$ of $\mathcal{F}_{\mathrm{GNN}}$ with

$$\log M \;\ge\; A_0/\epsilon^2, \qquad A_0 = C_A \frac{v_s^2 \log d}{L^2}.$$

Step 2: Fano's inequality. Let $Y = f^\star(X) + U$, with $U \sim \mathcal{N}(0, \sigma^2)$ i.i.d. For any two $f_j, f_k \in \mathcal{M}^*$, the corresponding distributions satisfy

$$\mathrm{KL}(P_j \| P_k) = \frac{\|f_j - f_k\|^2_{L_2(P_X)}}{2\sigma^2}.$$

Because the packing is constructed at radius 2ϵ, all pairs obey $\|f_j - f_k\|^2_{L_2} \ge (2\epsilon)^2$. To avoid degenerate constants, we further assume that separation does not exceed a constant factor, i.e.

$$\|f_j - f_k\|^2_{L_2} \le C_{\mathrm{KL}}\, \epsilon^2 \qquad \text{for some } C_{\mathrm{KL}} \ge 2.$$

(If all pairs are exactly 2ϵ-apart, then $C_{\mathrm{KL}} = 2$.) Thus $\mathrm{KL}_{\max} \le C_{\mathrm{KL}}\epsilon^2/\sigma^2$.

We apply the fixed-radius form of Fano's inequality (Lemma 2):

$$\mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;\ge\; \sup_{\epsilon>0} \left\{ \frac{\epsilon^2}{2} \left(1 - \frac{nC_{\mathrm{KL}}\epsilon^2/\sigma^2 + \log 2}{A_0/\epsilon^2} \right) \right\}.$$

Step 3: Optimizing over ϵ. Let $x = \epsilon^2$. The bound reads

$$g(x) \;=\; \frac{x}{2} \left(1 - \frac{nC_{\mathrm{KL}}x^2/\sigma^2 + x\log 2}{A_0} \right).$$

Maximizing $g(x)$ exactly requires solving a cubic. For a clean bound it suffices to choose x so that the parenthesis is $1/2$, i.e.

$$1 - \frac{nC_{\mathrm{KL}}x^2/\sigma^2 + x\log 2}{A_0} = \tfrac{1}{2}.$$

This yields the quadratic

$$\frac{nC_{\mathrm{KL}}}{\sigma^2}x^2 + (\log 2)x - \tfrac{A_0}{2} = 0,$$

whose positive root is given by

$$x = \epsilon^2 = \frac{\sigma^2}{2nC_{\mathrm{KL}}} \left(-\log 2 + \sqrt{(\log 2)^2 + \tfrac{2nA_0C_{\mathrm{KL}}}{\sigma^2}} \right).$$

For a detailed derivation, we provide the quadratic solution in Appendix B.1.

For this choice,

$$\mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;\ge\; \tfrac{1}{4}\,\epsilon^2.$$

Step 4: Asymptotics and constant. When n is large enough that $\frac{2nA_0C_{\mathrm{KL}}}{\sigma^2} \gg (\log 2)^2$, we expand the square root:

$$\epsilon^2 \;\approx\; \frac{\sigma}{\sqrt{2C_{\mathrm{KL}}}}\sqrt{\tfrac{A_0}{n}}.$$

Thus

$$\mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;\geq\; \frac{1}{4}\cdot\frac{\sigma}{\sqrt{2C_{\mathrm{KL}}}}\sqrt{\tfrac{A_0}{n}} = \Big(\tfrac{\sqrt{C_A}}{4\sqrt{2C_{\mathrm{KL}}}}\Big)\frac{\sigma v_s}{L}\sqrt{\tfrac{\log d}{n}}.$$

Define $K_{\text{new}} = \frac{\sqrt{C_A}}{4\sqrt{2C_{\mathrm{KL}}}} > 0$.

Step 5: Validity for all n. The exact root expression for ϵ^2 shows that the bound holds for all $n \geq 1$, not just asymptotically. Writing

$$\frac{\epsilon^2}{4} = \frac{\sigma^2}{8nC_{\mathrm{KL}}}\cdot\left(-\log 2 + \sqrt{(\log 2)^2 + \tfrac{2nA_0C_{\mathrm{KL}}}{\sigma^2}}\right),$$

one checks that the bracketed term is $\Omega(n^{1/2})$, hence the rate $K_{\text{new}}(\sigma v_s/L)\sqrt{(\log d)/n}$ holds uniformly in n (with a smaller constant if n is very small).

Step 6: Dimension condition. Finally, $d \geq 2$ ensures $\log d > 0$ so that $A_0 > 0$.

This completes the proof of Theorem 1. □

B.1 Exact Quadratic Solution for ϵ^2

In Step 3 of the proof of Theorem 1, we choose $\epsilon^2 = x$ so that the parenthetical term in Fano's bound equals $1/2$:

$$1 - \frac{nC_{\mathrm{KL}}x^2/\sigma^2 + x\log 2}{A_0} \;=\; \tfrac{1}{2}, \qquad A_0 \;=\; C_A\,\frac{v_s^2\log d}{L^2}.$$

This yields the quadratic

$$\frac{nC_{\mathrm{KL}}}{\sigma^2}\,x^2 \;+\; (\log 2)\,x \;-\; \frac{A_0}{2} \;=\; 0,$$

whose positive root is

$$\boxed{\epsilon^2 \;=\; x \;=\; \frac{\sigma^2}{2nC_{\mathrm{KL}}}\left(-\log 2 \;+\; \sqrt{(\log 2)^2 \;+\; \frac{2nA_0C_{\mathrm{KL}}}{\sigma^2}}\right).} \tag{20}$$

Substituting Eq. (20) into the fixed-radius Fano inequality (Lemma 2) gives

$$\mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;\geq\; \frac{\epsilon^2}{4} \;=\; \frac{\sigma^2}{8nC_{\mathrm{KL}}}\left(-\log 2 \;+\; \sqrt{(\log 2)^2 \;+\; \frac{2nA_0C_{\mathrm{KL}}}{\sigma^2}}\right). \tag{21}$$

Asymptotics. When n is large enough that $\frac{2nA_0C_{\mathrm{KL}}}{\sigma^2} \gg (\log 2)^2$, a first-order expansion of the square root in Eq. (20) gives

$$\epsilon^2 \;=\; \frac{\sigma}{\sqrt{2C_{\mathrm{KL}}}}\sqrt{\frac{A_0}{n}}\,\big(1{+}o(1)\big), \qquad \Rightarrow \qquad \mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;\geq\; \Big(\tfrac{\sqrt{C_A}}{4\sqrt{2C_{\mathrm{KL}}}}\Big)\frac{\sigma v_s}{L}\sqrt{\frac{\log d}{n}}\,(1{+}o(1)).$$

Uniform-in-n bound. Define

$$\Phi(n) \;:=\; -\log 2 \;+\; \sqrt{(\log 2)^2 \;+\; \frac{2nA_0C_{\mathrm{KL}}}{\sigma^2}}\,.$$

Then $\Phi(n)$ is strictly increasing in n, satisfies $\Phi(0) = 0$, and $\Phi(n) \sim \sqrt{2nA_0C_{\mathrm{KL}}}/\sigma$ as $n \to \infty$. From Eq. (21),

$$\mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;=\; \frac{\sigma^2}{8nC_{\mathrm{KL}}}\,\Phi(n) \;\geq\; \Big(\inf_{1\leq m\leq n_0}\frac{\sigma^2\,\Phi(m)}{8mC_{\mathrm{KL}}\,K_\star}\Big)\cdot K_\star\cdot\frac{1}{\sqrt{n}},$$

for any fixed $n_0 \in \mathbb{N}$ and target rate $K_\star := \frac{\sqrt{A_0}}{2}$. Choosing

$$K_{\text{new}} \;:=\; \min\left\{ \frac{\sqrt{C_A}}{4\sqrt{2C_{\mathrm{KL}}}},\;\; \min_{1\le m\le n_0} \frac{\sigma\,\Phi(m)}{4\sqrt{2C_{\mathrm{KL}}}\,\sqrt{mA_0}} \right\} > 0,$$

we obtain the uniform (in $n \ge 1$) lower bound

$$\mathcal{R}_{(n,|V|)}(\mathcal{F}_{\mathrm{GNN}}) \;\ge\; K_{\text{new}}\,\frac{\sigma v_s}{L}\sqrt{\frac{\log d}{n}}.$$

This shows the $\Omega\big((\sigma v_s/L)\sqrt{(\log d)/n}\big)$ rate holds for all $n \ge 1$ (with a possibly smaller K_{new} for very small n), while the asymptotic constant $\frac{\sqrt{C_A}}{4\sqrt{2C_{\mathrm{KL}}}}$ is recovered as $n \to \infty$.

Remark 3 (Why path graphs?)**.** *The path graph P_m minimizes connectivity and mixing: each node has degree at most two, and lazy random-walk mixing is slow, so one message-passing step propagates information only along a single chain. This bottlenecks information flow per layer, making depth the dominant factor. More connected graphs (e.g., expanders or dense random graphs) mix faster, which can only help learning. Hence, demonstrating hardness on path graphs suffices to certify a* minimax *lower bound for all admissible graphs—standard practice in worst-case lower-bound arguments.*

Remark 4 (Where topology enters the proof, and why a path)**.** *Graph topology influences the proof in two places:*

1. ***Packing construction.*** *Let $\mathcal{N}_1(v)$ denote the radius-1 neighborhood. We choose a set S of m nodes and vary only their first-layer weights. To avoid interference, we require $\{\mathcal{N}_1(v) : v \in S\}$ to be pairwise disjoint. On a path P_m this holds if the distance between consecutive nodes in S is at least 2, giving $|S| = \Theta(m)$. On a graph with maximum degree Δ, disjointness typically forces spacing $\ge \Delta{+}1$, reducing $|S|$ by a factor $\tilde{c}(\Delta) \le 1$ and thus shrinking the packing number by constants.*

2. ***KL–divergence control.*** *For Gaussian noise,*

$$\mathrm{KL}(P_j \| P_k) = \frac{\|f_j - f_k\|_{L_2}^2}{2\sigma^2} = \frac{1}{2\sigma^2}\sum_v (f_j(v) - f_k(v))^2.$$

With disjoint neighborhoods, a perturbation affects only outputs inside $\mathcal{N}_1(v)$. On P_m, $|\mathcal{N}_1(v)| \le 2$, so the KL scales like $O(|S|)$ for fixed perturbation size. On degree-Δ graphs, $|\mathcal{N}_1(v)| \le \Delta$, so for the same perturbation size the KL is larger by $O(\Delta)$. To keep KL bounded, we rescale the perturbation by $1/\sqrt{\Delta}$, which weakens the separation by the same factor. Both effects alter constants in Fano's inequality, not the n–dependence.

Consequence. *Because paths minimize degree ($\Delta = 2$) and maximize the number of disjoint radius-1 neighborhoods, they yield the tightest constants and the cleanest exposition. Moreover, any graph containing an induced path of length $\Omega(n)$ admits the same lower-bound* rate *as Theorem 1 (up to universal constants) by restricting the construction to that path.*

C Interpreting the Spectral–Homophily Assumption

Structural, not label-based. The assumption $\lambda_2(\mathcal{L}_n) \le \kappa/\log n$ concerns the spectrum of the normalized Laplacian of the subgraph induced by the n labeled nodes. It constrains *expansion and mixing* properties of the graph and is independent of labels or features. In particular, the condition can hold even if labels are adversarially assigned; no form of label homophily is required.

Why it makes learning harder. A small $\lambda_2(\mathcal{L}_n)$ implies low conductance and slow random-walk mixing by Cheeger-type inequalities (Bandeira et al., 2013). In this regime, message passing repeatedly reuses the same information: after $O(r)$ hops, neighborhoods overlap substantially. Our proof shows that $r = \Theta(\log n)$ suffices to reduce cross-block dependence below a fixed constant, so only $\Theta(\log n)$ blocks behave "nearly independently." This effective reduction in sample size yields the $\Omega(d/\log n)$ lower bound.

When the condition fails. Graphs with strong cross-cluster connectivity (i.e., good expansion) typically have λ_2 bounded away from 0 (often $\Theta(1)$). Such graphs fall outside the assumption, and the guarantee reverts to the $\Omega\big(\sqrt{\log d/n}\big)$ rate of Theorem 1.

Examples.

- *Paths, cycles, or chain-of-cliques:* $\lambda_2(\mathcal{L})$ decays with graph size. For sufficiently large n, the condition $\lambda_2 \le \kappa/\log n$ is satisfied, often by a wide margin.
- *Expanders and dense random graphs:* $\lambda_2(\mathcal{L}) = \Theta(1)$, so the condition fails and the analysis falls back to Theorem 1.

D Fano's Inequality (Fixed-Radius Form)

We state the specific version of Fano's inequality used throughout the proofs. It is a standard corollary of Lemma 2.10 in Tsybakov (2009).

Lemma 2 (Fano–Tsybakov, fixed-radius form)**.** *Let (Θ, d) be a metric space, and let $\{\mathbb{P}_\theta : \theta \in \Theta\}$ be a family of distributions on $\mathcal{X}$. Suppose there exist $M \ge 2$ points $\theta_1, \dots, \theta_M \in \Theta$ such that:*

(i) ***Separation:*** *$d(\theta_j, \theta_k) \ge 2\varepsilon$ for all $j \ne k$;*

(ii) ***KL control:*** *$\max_{j \ne k} \mathrm{KL}(\mathbb{P}_{\theta_j} \,\|\, \mathbb{P}_{\theta_k}) \le \beta$.*

Then, for any estimator $\hat{\theta}$,

$$\inf_{\hat{\theta}} \sup_{\theta \in \{\theta_1, \dots, \theta_M\}} \mathbb{E}_\theta\big[d(\hat{\theta}, \theta)^2\big] \;\ge\; \frac{\varepsilon^2}{2}\left(1 - \frac{\beta + \log 2}{\log M}\right).$$

The bound is meaningful whenever $\beta \le \frac{1}{2}\log M - \log 2$.

This fixed-radius form is the one applied in all lower-bound arguments. It follows directly from Lemma 2.10 in Tsybakov (2009), but is stated here for completeness and to keep the paper self-contained.

E Mixing Time and Spectral Gap

F Mixing Time and the Spectral Gap

We formally justify that the spectral–homophily condition in Theorem 2 implies logarithmic random-walk mixing time.

Lemma 3 (Mixing time via spectral gap)**.** *Let $G = (V, E)$ be a finite, connected, undirected graph, and let $P = \frac{1}{2}I + \frac{1}{2}D^{-1}A$ be the lazy random-walk transition matrix, where A is the adjacency matrix and $D = \mathrm{diag}(\deg(v))$. The stationary distribution is*

$$\pi(v) = \frac{\deg(v)}{2|E|}, \qquad v \in V,$$

so that

$$\pi_{\min} \;\ge\; \frac{1}{2|E|} \;\ge\; \frac{1}{|V|^2}.$$

For every $\varepsilon \in (0, 1)$,

$$t_{\mathrm{mix}}(\varepsilon) \;\le\; \frac{\log\big(1/(\varepsilon\pi_{\min})\big)}{1 - \lambda_2} \;\le\; \frac{2\log|V| + \log(1/\varepsilon)}{1 - \lambda_2},$$

where λ_2 is the second largest eigenvalue of P (the spectral gap is $1 - \lambda_2 > 0$). (Levin & Peres, 2017, Theorem 12.4, (12.10)).

Proof. By reversibility of P, the stationary distribution is $\pi(v) = \deg(v)/(2|E|)$. Hence

$$\pi_{\min} = \min_v \pi(v) = \frac{\deg_{\min}}{2|E|} \;\geq\; \frac{1}{2|E|} \;\geq\; \frac{1}{|V|^2},$$

since $|E| \leq |V|(|V|-1)/2$.

Let $\lambda_2 = \lambda_2(P)$ denote the second largest eigenvalue. Standard spectral bounds for lazy reversible chains (Levin & Peres, 2017, Theorem 12.4) yield

$$t_{\text{mix}}(\varepsilon) \;\leq\; \frac{\log\big(1/(\varepsilon\pi_{\min})\big)}{1-\lambda_2}, \qquad \varepsilon \in (0,1).$$

Substituting the bound on $\pi_{\min}$ gives

$$\log\Big(\tfrac{1}{\varepsilon\pi_{\min}}\Big) \;\leq\; \log\Big(\tfrac{1}{\varepsilon}\Big) + 2\log|V|.$$

Thus

$$t_{\text{mix}}(\varepsilon) \;\leq\; \frac{2\log|V| + \log(1/\varepsilon)}{1-\lambda_2}.$$

If $\lambda_2 \leq \kappa/\log n$ with $n = |V|$ and fixed κ, then for sufficiently large n we have $1-\lambda_2 \geq 1-\kappa/\log n \geq c$ for some universal $c \in (0,1)$. Hence $t_{\text{mix}}(\varepsilon) = O(\log n)$. □

Implication. The bound implies that a mixing radius of order $r_{\text{mix}} = \Theta(\log n)$ suffices. Consequently, we may select $K(\lambda_2, \kappa) = \Theta(\log n)$ nodes whose neighborhoods can be treated as effectively disjoint in our hypothesis construction. This is encoded through the constant $c_{\text{mix}}(\lambda_2, \kappa)$ used in Section 4.

F.1 Lower Bound on the Stationary Distribution

For completeness, we justify the lower bound on $\pi_{\min}$ used above. Since $\pi(v) = \deg(v)/(2|E|)$ for the lazy walk,

$$\pi_{\min} = \frac{\deg_{\min}}{2|E|} \;\geq\; \frac{1}{2|E|}.$$

As $|E| \leq |V|(|V|-1)/2$, it follows that

$$\pi_{\min} \;\geq\; \frac{1}{|V|^2}.$$

This universal bound is adopted in Lemma 3. Sharper bounds (e.g., $\pi_{\min} \geq c/|V|$) require minimum-degree assumptions such as $\deg_{\min} \geq c|V|$, which we do not impose here. Our rates therefore conservatively rely on the $1/|V|^2$ bound.

G Refining Γ in the Lower Bound

As shown in the main proof in Section 4, the minimax risk is lower bounded by a rate of $\frac{\sigma^2 v_s^2 d}{L^2 \log n}$. This appendix refines the constant $\Gamma = \Gamma_c/(2c_2)$ in that bound by specifying sufficient conditions under which the inequality in Eq. (18) is satisfied. The condition was $c_1 K\sigma^2 L^2 K \geq \Delta_H v_s^2 + 2\sigma^2 L^2 K \log 2$. Substituting $\Delta_H \approx \frac{16\sigma^2 d}{\Gamma_c}$:

$$c_1 K^2\sigma^2 L^2 \geq \frac{16\sigma^2 d}{\Gamma_c} v_s^2 + 2\sigma^2 L^2 K \log 2 \quad\Longrightarrow\quad c_1 K^2 L^2 \geq \frac{16 d v_s^2}{\Gamma_c} + 2L^2 K \log 2.$$

This condition essentially requires that $\frac{16dv_s^2}{\Gamma_c}$ is not too large compared to $K^2L^2 = (\log n)^2 L^2$. Specifically, we need $\Gamma_c \geq \frac{16dv_s^2}{(c_1 K^2 L^2 - 2L^2 K \log 2)}$. For large n, we can approximate this as $\Gamma_c \gtrsim \frac{dv_s^2}{K^2L^2} = \frac{dv_s^2}{(\log n)^2 L^2}$. We also need $\Gamma_c > 64\sigma^2$ (for $\Delta_H \leq d/4$). So, Γ_c must be chosen as a sufficiently large universal constant, potentially depending on fixed universal constants like c_1 and desired Fano factor c_2, and satisfying these conditions. If $dv_s^2/((\log n)^2 L^2)$ is bounded by a constant (which is often an implicit assumption on how d can scale with n for the bound to be non-trivial or for the construction to be valid), then Γ_c can be chosen as a constant. The resulting $\Gamma = \Gamma_c/(2c_2)$ is then a universal constant, depending on properties of the function class (implicitly through L, v_s in the conditions for Γ_c) and the packing construction (through c_1, c_2).

H CONSTANT DEFINITIONS AND EXPLICIT-K FORM

Definition of Γ and its dependencies. Γ collects the universal constants that arise in the Fano argument: (i) the packing/code-size constant from the Varshamov–Gilbert construction (c_{VG}), (ii) the constant in the upper bound on the KL divergence between hypotheses (c_{KL}), and (iii) the slack constant from Fano's inequality (c_2). After Step 3 of the proof (Section 4), the bound becomes

$$\mathcal{R}^{\mathrm{node}}_{(n,G)}(\mathcal{F}_{\mathrm{GNN}}) \;\geq\; \frac{\sigma^2 v_s^2}{L^2} \cdot \frac{d}{K(\lambda_2,\kappa)} \cdot \frac{c_{\mathrm{VG}}}{16\, c_{\mathrm{KL}}} \cdot \frac{1}{2c_2},$$

where $K(\lambda_2,\kappa)$ denotes the effective number of nearly independent blocks obtained from the mixing-time argument. Thus, we may define

$$\Gamma \;:=\; \Gamma_c\,(2c_2) \quad \text{with} \quad \Gamma_c := \frac{16\, c_{\mathrm{KL}}}{c_{\mathrm{VG}}}.$$

These constants depend only on the geometry of the function class, as determined by the packing/separation construction (ReLU Lipschitz constant implied by the ℓ_1-budget v_s, the depth L, and the orthogonality of features used in the packing), and on the fixed inequalities invoked in the proof. Importantly, Γ is independent of n and d, apart from the explicit factors already shown in the bound.

Equivalent statement with explicit spectral dependence. With this notation, the bound can be written as

$$\mathcal{R}^{\mathrm{node}}_{(n,G)}(\mathcal{F}_{\mathrm{GNN}}) \;\geq\; \frac{\sigma^2 v_s^2}{\Gamma' L^2} \cdot \frac{d}{K(\lambda_2,\kappa)}, \tag{22}$$

for a universal constant $\Gamma' > 0$ (absorbing the universal constants $c_{\mathrm{VG}}, c_{\mathrm{KL}}, c_2$). Under the spectral–homophily condition $\lambda_2 \leq \kappa/\log n$, one has $K(\lambda_2,\kappa) = \Theta(\log n)$, and Eq. 22 reduces to Theorem 2.

Remark 5 (On the role of κ and λ_2)**.** *The parameters κ and λ_2 enter through $K(\lambda_2,\kappa)$, the effective count of "independent blocks" provided by the mixing argument. Once we substitute $K(\lambda_2,\kappa) = \Theta(\log n)$ under $\lambda_2 \leq \kappa/\log n$, their influence reduces to a constant multiplicative factor, which is absorbed into Γ.*

I OPERATOR-NORM CONTROL FOR ADJACENCY-MASKED ATTENTION

This section provides the operator-norm analysis that underpins the applicability of Theorem 2 to adjacency-masked attention layers. For the complementary GAT-specific discussion, see Appendix J.

Conditions for applicability. Theorem 2 extends to attention-based GNNs provided the following hold: (i) *Adjacency masking:* each head attends only to $\mathcal{N}(i)$ (or, more generally, an r-hop neighborhood); (ii) *Bounded layer operators:* each layer is Lipschitz with uniformly bounded operator norm (e.g., via bounding attention scores by temperature/clipping or constraining the attention matrix norm); (iii) *Finite depth L.* Under (i)–(iii), the proof is unchanged up to constants depending on the product of layer norms and, for r-hop masking, on r. Fully global (unmasked) attention is non-local and therefore outside the locality premise of Theorem 2.

Norm-control derivation. Consider a single masked attention head with queries $Q = HW_Q$, keys $K = HW_K$, values $V = HW_V$, and adjacency mask $M \in \{0, -\infty\}^{|V|\times|V|}$ restricting attention to $\mathcal{N}(i)$ (or an r-hop pattern). With temperature $\tau > 0$ and row-wise softmax,

$$A = \mathrm{softmax}\big((QK^\top + M)/\tau\big),$$

and the layer map is $H \mapsto AV$ (plus a 1×1 mixing which we absorb into the operator norm). Assume $\|W_Q\|_2 \leq c_Q$, $\|W_K\|_2 \leq c_K$, $\|W_V\|_2 \leq c_V$, and rows of Q, K are bounded in norm by B (this holds if $\|H\|_F$ is controlled inductively and layer norms are bounded). Then each masked row of $(QK^\top)/\tau$ has entries bounded by $B^2 c_Q c_K/\tau$, so the softmax is α-Lipschitz on each row with $\alpha \leq C_\tau$ and yields a row-stochastic A supported on the mask. Hence $\|A\|_2 \leq 1$ and

$$\|AV\|_2 \;\leq\; \|A\|_2\, \|V\|_2 \;\leq\; c_V\, \|H\|_2.$$

With residual/linear projections folded in, the per-layer Lipschitz constant is bounded by a product of operator norms (one per submodule), yielding a uniform bound $L_{\mathrm{op}} < \infty$ per layer. Therefore a depth-L masked-attention stack has overall Lipschitz constant $\leq (L_{\mathrm{op}})^L$. The proof of Theorem 2 uses only: (a) adjacency locality from masking, and (b) bounded layer Lipschitz constants. Both hold under the above conditions, so the same packing, KL control, and Fano steps go through with constants depending on L_{op} (and on r for r-hop masks).

J Adjacency-Masked GAT Layers under Theorem 2

This section explains why standard GAT layers fit within the assumptions of Theorem 2. For the accompanying operator-norm control argument, see Appendix I.

Theorem 1 assumes (A1) and thus *excludes* input-dependent mixing (attention). By contrast, Theorem 2 requires only adjacency-masked 1-hop receptive fields and bounded operator norms. Standard GAT layers satisfy these conditions if attention is restricted to $\mathcal{N}(i)$ and softmax weights are bounded (e.g., via temperature or clipping).

Formally, a single GAT layer with adjacency mask can be written as

$$h_i^{(\ell+1)} \;=\; \phi\left(W^{(\ell)} \sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(\ell)}(H^{(\ell)})\, h_j^{(\ell)} \;+\; B^{(\ell)} h_i^{(\ell)} \right),$$

where $\sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(\ell)}(\cdot) = 1$, $\alpha_{ij}^{(\ell)} \geq 0$, and $\alpha_{ij}^{(\ell)} = 0$ for $j \notin \mathcal{N}(i)$. Although the coefficients depend on features (violating (A1)), they are *adjacency-masked* and convex.

Assume (i) the attention logits are bounded (e.g., softmax with temperature or clipping), so that $\max_i \sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(\ell)} \leq C_{\mathrm{att}}$ and the Jacobian of the mapping $H^{(\ell)} \mapsto \{\alpha_{ij}^{(\ell)}\}$ is bounded; and (ii) the linear maps satisfy the same ℓ_1 budget as in (A2). Then the layer is Lipschitz with operator-norm bound $\|W^{(\ell)}\| \cdot C_{\mathrm{att}} + \|B^{(\ell)}\|$ (with the dependence on the attention logits' temperature absorbed into C_{att}).

The proof of Theorem 2 uses only: (a) adjacency locality (receptive field confined to the graph), (b) bounded layer Lipschitz constants, and (c) the graph mixing argument yielding $K = \Theta(\log n)$ effectively independent blocks under $\lambda_2(\mathcal{L}) \leq \kappa/\log n$. Conditions (a)–(b) hold for adjacency-masked GAT with bounded logits, hence the same packing, KL control, and Fano steps go through with the constants absorbed into Γ. Therefore, the $\Omega\big(d/\log n\big)$ lower bound applies to standard GAT under these mild norm constraints. In contrast, Theorem 1 explicitly relies on input-independent aggregation and does not cover attention.

K Practical guidance for data-scarce graphs

The structure-aware lower bound (Theorem 2) implies that when only $\tilde{O}(\log n)$ training nodes are effectively independent, naive data scaling is statistically inefficient. Constants in the bound can often be improved in practice, though the asymptotic rate $\Omega(d/\log n)$ remains unchanged. The following interventions help improve constants:

- **Break neighborhood homogeneity / slow mixing.** Add node individualization or positional encodings (e.g., random/learned IDs, Laplacian/RW features) and consider heterophily-aware layers; these reduce overlap of message-passing neighborhoods.
- **Reduce effective dimension before fine-tuning.** Use transfer or self-supervised pretraining on large auxiliary graphs, then freeze most layers or select features to shrink the effective d entering the bound.
- **Diversify supervision.** Active/coreset label selection that spreads labels across loosely connected regions (far in graph distance or across communities) increases independence among samples.
- **Regularize against slow mixing / over-smoothing.** Use residual/JK connections, PPR/teleport propagation, DropEdge/edge sparsification, and limit depth; these shorten the mixing horizon, raising the usable information per label.

Takeaway. These choices increase the informative signal per labeled node and improve constants in $\frac{\sigma^2 v_s^2}{\Gamma L^2} \cdot \frac{d}{\log n}$, but the qualitative $\log n$ denominator remains the limiting factor under the spectral–homophily condition.

L Architectural drivers of heterogeneous scaling

Why different models show different scaling on the *same* dataset. Even on a fixed graph, architectures can induce different effective sample efficiencies due to variation in receptive-field growth and information reuse. We identify two main drivers:

- **Smoothing and receptive-field growth.** GCN's fixed, normalized adjacency (a graph-dependent but input-independent filter) resembles a classical spectral filter. When the task's signal is spectrally aligned, this can yield faster apparent decay (closer to $1/\sqrt{n}$). By contrast, GAT and GraphSAGE adapt mixing weights and thereby emphasize homophilous neighborhoods; this adaptation increases overlap among message-passing neighborhoods and reduces the effective number of independent samples, exposing the slower $1/\log n$ decay predicted by Theorem 2.
- **Bias–variance tradeoffs and noise floors.** Models with stronger inductive bias (e.g., GCN) can reach a bias-dominated error floor early, which makes the observed asymptotic slope appear steeper. More flexible models (GAT/GraphSAGE) reduce bias but incur higher variance, which dissipates slowly because samples are not effectively independent under overlapping neighborhoods.

This perspective helps explain the heterogeneous scaling observed in Table 2 (e.g., GCN on Reddit favoring $1/\sqrt{n}$, versus GAT and GraphSAGE favoring $1/\log n$).

M Experimental Details and Settings

This appendix details all elements of the experimental setup, training configuration, evaluation protocols, model fitting, and resource usage to ensure reproducibility of our results.

Environment and Compute Resources

All experiments were conducted using `PyTorch` and `PyTorch Geometric` (PyG). We used a GPU-enabled machine equipped with an NVIDIA Tesla V100 (32GB VRAM) and 64GB system RAM. Each experiment (one training size n with one model on one dataset) typically completed in under 5 minutes for smaller $n \leq 1000$, and under 15 minutes for large-scale datasets like Reddit and QM9 with $n = 50{,}000$. The total compute budget, including all training, evaluation, curve fitting, and figure/table generation, was under 50 GPU-hours.

Dataset Licenses and Citations

The following publicly available datasets were used in this study, all accessed through the `torch_geometric.datasets` module. Below we provide license information and cite the original sources in accordance with reproducibility and usage guidelines.

- **Cora** (McCallum et al., 2000): A citation network with $|V| = 2{,}708$ nodes and $|E| = 5{,}429$ edges, used for node classification (error rate). Available via the LINQS dataset repository: `https://linqs.org/datasets/`. No license was explicitly stated in the original publication.
- **Reddit** (Hamilton et al., 2017): A large-scale social network with $|V| = 232{,}965$ nodes and $|E| = 11{,}606{,}948$ edges, used for community detection (error rate). The dataset is derived from Reddit data and is subject to Reddit's API terms of service.
- **QM9** (Ramakrishnan et al., 2014b): A molecular graph dataset with average $|V| \approx 18$ nodes and $|E| \approx 40$ edges, used for graph-level regression (mean squared error, MSE). Licensed

under Creative Commons Attribution 4.0 International (CC BY 4.0) and available at `https://doi.org/10.6084/m9.figshare.c.978904.v5`.

- **Facebook (Page-Page)** (Rozemberczki et al., 2019): A page-to-page graph from Facebook with $|V| = 4{,}039$ nodes and $|E| = 88{,}234$ edges, used for link prediction (1-AUC). The dataset is distributed under a Creative Commons Attribution 4.0 International (CC BY 4.0) license and can be accessed via `https://snap.stanford.edu/data/ego-Facebook.html`.

Models and Architecture

We evaluated the following Graph Neural Network (GNN) architectures:

- **GCN**: 2-layer Graph Convolutional Network using `GCNConv`, with 16 hidden units.
- **GAT**: 2-layer Graph Attention Network with 8 heads in the first layer, and a single head in the second.
- **GraphSAGE**: 2-layer GraphSAGE model using `SAGEConv`, with 16 hidden units.

All models use ReLU activation after the first layer.

Tasks and Loss Functions

We tested three standard graph learning tasks:

- **Node Classification**: Cross-entropy loss on node labels.
- **Link Prediction**: Binary classification using the inner product decoder and binary cross-entropy with logits.
- **Graph Regression**: Molecular property prediction using mean squared error (MSE) on the target scalar field.

Training Protocol

- **Subset Sampling**: For each experiment, a subset of $n \in \{100, 500, 1k, 2.5k, 5k, 10k, 50k\}$ samples was randomly selected. For node and link tasks, subgraphs were constructed using `torch_geometric.utils.subgraph`.
- **Data Splits**: A fixed 80%/20% train/test split was used.
- **Optimizer**: Adam with learning rate 0.01, weight decay 10^{-4}.
- **Epochs**: 200.
- **Batch Size**: 32 for all tasks.
- **Evaluation Metrics**:
  - Misclassification rate for classification,
  - MSE for regression,
  - $1 - \text{AUC}$ for link prediction.

Statistical Significance and Error Reporting

Each experiment (fixed dataset, model, and n) was repeated 5 times with different random seeds. The reported error metric includes the sample mean and standard deviation across the 5 runs. Standard deviation is used for error bars and in weighted fitting procedures. These represent variation due to random sampling and initialization. All error bars shown in figures correspond to ± 1 standard deviation.

CURVE FITTING AND LEARNING TREND ANALYSIS

To analyze sample complexity trends, we fit the test error curves to the following models:

$$\text{Model 1:} \quad c_1 + \frac{\alpha}{\sqrt{n}}$$

$$\text{Model 2:} \quad c_2 + \frac{\beta}{n}$$

$$\text{Model 3:} \quad c_3 + \frac{\delta}{\log n}$$

$$\text{Model 4:} \quad c_4 + \frac{1}{n^\gamma}$$

Fits were performed using weighted least squares with weights $w_i = 1/\sigma_i^2$, where σ_i is the standard deviation of the ith data point. The power-law model was fitted using `scipy.optimize.curve_fit` with bounded parameters and a robust initial guess. For each model, we computed:

- Weighted Residual Sum of Squares (RSS)
- Weighted Mean Squared Error (MSE)
- Weighted R^2 value

The best fitting model for each dataset and architecture was determined based on the maximum R^2. Fitted parameters and metrics were summarized in a LaTeX-formatted table (`final_comparison_table_weighted.tex`), and model-specific figures were saved as `<dataset>_<model>_fits.png`.

VISUALIZATION AND REPRODUCIBILITY ASSETS

All figures include error bars, and each plot overlays all fitted models for comparison. All code, including data loading, model training, evaluation, fitting, table generation, and visualization, is structured and commented for reproducibility.

CODE AND REPRODUCIBILITY

To support verification and reproducibility, we provide the full source code as supplementary material. This includes implementations for data loading, model training, evaluation, error analysis, and curve fitting, as well as scripts to reproduce all experimental results, generate LaTeX tables, and visualize learning curves in line with reproducibility guidelines.

Summary: Every step necessary to replicate our results—datasets, architectures, parameters, training and evaluation setup, fitting strategy, and visualizations—is fully disclosed and executable by third parties with access to the same datasets and a standard GPU-enabled Python environment.

M.1 STRUCTURAL STATISTICS

To connect the empirical analysis with our theoretical results, we compute two structural measures for each dataset.

Homophily is defined as

$$h(G) = \frac{1}{|E|} \sum_{(u,v)\in E} \mathbf{1}\{y_u = y_v\},$$

where E is the edge set and y_u denotes the ground-truth label of node u.

Spectral gap. We compute $\lambda_2(\mathcal{L}_n)$, the second-smallest eigenvalue of the normalized Laplacian

$$\mathcal{L}_n = I - D_n^{-1/2} A_n D_n^{-1/2},$$

where A_n and D_n are the adjacency and degree matrices of the induced subgraph on labeled nodes. Both measures are derived directly from the observed graph and label information, ensuring consistency with the conditions stated in Theorem 2.

N THE USE OF LARGE LANGUAGE MODELS (LLMS)

In preparing this manuscript, we used Large Language Models (LLMs) solely as general-purpose assistive tools for grammar checking, language polishing, and improving clarity of exposition. LLMs were not used for research ideation, theoretical development, experiment design, or analysis, and they did not contribute any scientific content. The authors take full responsibility for the contents of the paper, including any parts where LLMs were used to improve writing style.