Graph Attention Retrospective

Anonymous authors
Paper under double-blind review

Abstract

Graph-based learning is a rapidly growing sub-field of machine learning with applications in social networks, citation networks, and bioinformatics. One of the most popular type of models is graph attention networks. These models were introduced to allow a node to aggregate information from the features of neighbor nodes in a non-uniform way in contrast to simple graph convolution which does not distinguish the neighbors of a node. In this paper, we study theoretically this expected behaviour of graph attention networks. We prove multiple results on the performance of the graph attention mechanism for the problem of node classification for a contextual stochastic block model. Here the features of the nodes are obtained from a mixture of Gaussians and the edges from a stochastic block model where the features and the edges are coupled in a natural way. First, we show that in an "easy" regime, where the distance between the means of the Gaussians is large enough, graph attention is able to distinguish inter-class from intra-class edges, and thus it maintains the weights of important edges and significantly reduces the weights of unimportant edges. As a corollary, we show that this implies perfect node classification. However, a classical argument shows that in the "easy" regime, the graph is not needed at all to classify the data with high probability. In the "hard" regime, we show that *every* attention mechanism fails to distinguish intra-class from inter-class edges. We evaluate our theoretical results on synthetic and real-world data.

1 Introduction

Graph learning has received a lot of attention recently due to breakthrough learning models (Gori et al., 2005; Scarselli et al., 2009; Bruna et al., 2014; Duvenaud et al., 2015; Henaff et al., 2015; Atwood & Towsley, 2016; Defferrard et al., 2016; Hamilton et al., 2017; Kipf & Welling, 2017) that are able to exploit multi-modal data which consist of nodes and their edges as well as the features of the nodes. One of the most important problems in graph learning is the problem of *classification*, where the goal is to classify the nodes or edges of a graph given the graph and the features of the nodes. Two of the most popular mechanisms for classification and graph learning in general are the graph convolution and the graph attention. Graph convolution, usually defined using its spatial version, corresponds to averaging the features of a node with the features of its neighbors (Kipf & Welling, 2017).[1] Graph attention (Velickovic et al., 2018) mechanisms augment this convolution by appropriately weighting the edges of a graph before spatially convolving the data. Graph attention is able to do this by using information from the given features for each node. Despite its wide adoption by practitioners (Fey & Lenssen, 2019; Wang et al., 2019; Hu et al., 2020) and its large academic impact as well, the number of works that rigorously study its effectiveness is quite limited.

One of the motivations for using a graph attention mechanism as opposed to a simple convolution is the expectation that the attention mechanism is able to distinguish inter-class edges from intra-class edges, and consequently weights inter-class edges and intra-class edges differently before performing the convolution step. This ability essentially maintains the weights of important edges and significantly reduces the weights of unimportant edges, and thus it allows graph convolution to aggregate features from a subset of neighbor nodes that would help node classificaiton tasks. In this work we explore the regimes in which this heuristic picture holds in simple node classification tasks, namely classifying the nodes in a contextual stochastic block model (CSBM) (Binkiewicz et al., 2017; Deshpande et al., 2018). The CSBM is a coupling of the stochastic block model (SBM) with a Gaussian mixture model, where the features of the nodes within a class are drawn from the same component of the mixture model. For a more precise definition, see Section 2. We focus on the case of two classes where the answer to the above question is sufficiently

[1] Although the model in Kipf & Welling (2017) is related to spectral convolutions, it is mainly a spatial convolution since messages are propagated along graph edges.

precise to understand the performance of graph attention and build useful intuition about it. We briefly and informally summarize our contributions as follows:

1. In the "easy regime", i.e., when the distance between the means is much larger than the standard deviation, we show that there exists a choice of attention architecture that distinguishes inter-class edges from intra-class edges with high probability. In particular, we show that the attention coefficients for one class of edges are much higher than the other class of edges. Furthermore, we show that these attention coefficients lead to perfect node classification result. However, in the same regime, we show that the graph is not needed to perfectly classify the data.
2. In the "hard regime", i.e., when the distance between the means is small compared to the standard deviation, we show that *any* attention architecture is unable to distinguish inter-class from intra-class edges with high probability. Moreover, we show that using the original GAT architecture (Velickovic et al., 2018), with high probability, most of the attention coefficients are going to have uniform weights, similar to those of uniform graph convolution (Kipf & Welling, 2017).
3. We provide an extensive set of experiments both on synthetic data, and on three popular real-world datasets that validates our theoretical results.

1.1 Previous work

Recently the concept of attention for neural networks (Bahdanau et al., 2015; Vaswani et al., 2017) was transferred to graph neural networks (Li et al., 2016; Bresson & Laurent, 2018; Velickovic et al., 2018; Lee et al., 2019; Puny et al., 2020). A few papers have attempted to understand the mechanism in Velickovic et al. (2018). One work relevant to ours is Brody et al. (2022). In this paper the authors show that a node may fail to assign large edge weight to its most important neighbors due to a global ranking of nodes that is generated by the attention mechanism in Velickovic et al. (2018). Another related work is Knyazev et al. (2019), which presents an empirical study of the ability of graph attention to generalize on larger, complex, and noisy graphs. In addition, in Hou et al. (2019) the authors propose a different metric to generate the attention coefficients and show empirically that it has an advantage over the original GAT architecture. Other related work to ours, which does not focus on graph attention, comes from the field of statistical learning on random data models. In particular, random graphs and the stochastic block model have been traditionally used in clustering and community detection (Abbe, 2018; Athreya et al., 2018; Moore, 2017). Moreover, the works by Binkiewicz et al. (2017); Deshpande et al. (2018), which also rely on CSBM are focused on the fundamental limits of unsupervised learning. Of particular relevance is the work by Baranwal et al. (2021), which studies the performance of graph convolution on CSBM as a semi-supervised learning problem. Finally, there are a few related theoretical works on understanding the performance and the universality of graph neural networks (Chen et al., 2019; Chien et al., 2021; Zhu et al., 2020; Xu et al., 2019; Garg et al., 2020; Loukas, 2020a;b). We provide theoretical results that characterize the precise performance of graph attention compared to graph convolution and no convolution for CSBM with the goal of answering the particular questions that we imposed above.

2 Preliminaries

In this section, we describe the *Contextual Stochastic Block Model (CSBM)* (Deshpande et al., 2018) which serves as our data model, and the *Graph Attention* mechanism (Velickovic et al., 2018).

Let $d, n \in \mathbb{N}$, and $\epsilon_1, \dots, \epsilon_n \sim \text{Ber}(1/2)$[2], and define two classes as $C_k = \{j \in [n] \mid \epsilon_j = k\}$ for $k \in \{0, 1\}$. For each index $i \in [n]$, we set the feature vector $\mathbf{X}_i \in \mathbb{R}^d$ as $\mathbf{X}_i \sim N((2\epsilon_i - 1)\boldsymbol{\mu}, \mathbf{I} \cdot \sigma^2)$[3] where $\boldsymbol{\mu} \in \mathbb{R}^d$, $\sigma \in \mathbb{R}$ and $\mathbf{I} \in \{0, 1\}^{d \times d}$ is the identity matrix. For a given pair $p, q \in [0, 1]$ we consider the stochastic adjacency matrix $\mathbf{A} \in \{0, 1\}^{n \times n}$ defined as follows. For $i, j \in [n]$ in the same class (i.e., *intra-class edge* or simply *intra-edge*), we set $a_{ij} \sim \text{Ber}(p)$, and if i, j are in different classes (i.e., *inter-class edge* or simply *inter-edge*), we set $a_{ij} \sim \text{Ber}(q)$. We denote by $(\mathbf{X}, \mathbf{A}) \sim \mathsf{CSBM}(n, p, q, \boldsymbol{\mu}, \sigma^2)$ a sample obtained according to the above random process. An advantage of CSBM is that it allows us to control the noise by controlling the parameters of the distributions of the model. In particular, CSBM allows us to control the distance of the means and the variance of the Gaussians, which are important for controlling separability of the Gaussians. For example, fixing the variance, then the closer the means are the more

[2] $\text{Ber}(\cdot)$ denotes the Bernoulli distribution.

[3] The means of the mixture of Gaussians are $\pm\boldsymbol{\mu}$. Our results can be easily generalized to general means. The current setting makes our analysis simpler without loss of generality.

difficult the separation of the Gaussians becomes. Moreover, CSBM allows us to control the noise in the graph, namely the difference between intra-class and inter-class edge probabilities.

A *single-head* graph attention applies some weight function on the edges based on their node features (or a mapping thereof). Given two representations $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^{F'}$ for two nodes $i, j \in [n]$, let $\Psi(\boldsymbol{h}_i, \boldsymbol{h}_j) \overset{\text{def}}{=} \alpha(\mathbf{W}\boldsymbol{h}_i, \mathbf{W}\boldsymbol{h}_j)$ where $\alpha : \mathbb{R}^F \times \mathbb{R}^F \to \mathbb{R}$ and $\mathbf{W} \in \mathbb{R}^{F \times F'}$ is a learnable matrix. We refer to the mapping Ψ as the *attention model/mechanism* with *attention coefficients*:

$$\gamma_{ij} \overset{\text{def}}{=} \frac{\exp(\Psi(\boldsymbol{h}_i, \boldsymbol{h}_j))}{\sum_{\ell \in N_i} \exp(\Psi(\boldsymbol{h}_i, \boldsymbol{h}_\ell))}, \tag{1}$$

where N_i is the set of neighbors of node i including node i itself. Letting f be some nonlinear element-wise function, the graph attention convolution output is $\tilde{\boldsymbol{h}}_i = f(\boldsymbol{h}'_i)$, where $\boldsymbol{h}'_i = \sum_{j \in [n]} \mathbf{A}_{ij} \gamma_{ij} \mathbf{W} \boldsymbol{h}_j \quad \forall i \in [n]$. A *multi-head* graph attention (Velickovic et al., 2018) uses $K \in \mathbb{N}$ weight matrices $\mathbf{W}^1, \dots, \mathbf{W}^K \in \mathbb{R}^{F \times F'}$ and averages their individual (single-head) outputs. We consider the most simplified case of a single graph attention layer (i.e., $F' = d$ and $F = 1$) where α is realized by an MLP using LeakyRelu activation.[4]

The CSBM model induces dataset features $\mathbf{X}$ which are correlated through the graph $G = ([n], E)$, represented by an adjacency matrix $\mathbf{A}$. A natural requirement of an attention architecture is to maintain important edges in the graph and ignore unimportant edges. For example, important edges could be the set of intra-class edges and unimportant edges could be the set of inter-class edges. In this case, if graph attention maintains all intra-class edges and ignores all inter-class edges, then a node from a class will be connected only to nodes from its own class. More specifically, a node v will be connected to neighbor nodes whose associated node features come from the *same distribution* as node features of v. Given two sets A and B, we denote $A \times B \overset{\text{def}}{=} \{(i,j) : i \in A, j \in B\}$ and $A^2 \overset{\text{def}}{=} A \times A$. To study the expected behavior that graph attention should main important edges and drop unimportant edges, we use the following definition of separability of edges. Given an attention model Ψ, we say that the model *separates the edges*, if the outputs $\Psi(\mathbf{X}_i, \mathbf{X}_j)$ satisfy $\text{sign}(\Psi(\mathbf{X}_i, \mathbf{X}_j)) = \text{sign}(p - q)$ when (i, j) is an intra-class edge, i.e. $(i, j) \in (C_1^2 \cup C_0^2) \cap E$, and $\text{sign}(\Psi(\mathbf{X}_i, \mathbf{X}_j)) = \text{sign}(q - p)$ when (i, j) is an inter-class edge, i.e. $(i, j) \in E \setminus (C_1^2 \cup C_0^2)$.[5]

3 Results

We consider two parameter regimes: the first ("easy regime") is where $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$, and the second ("hard regime") is where $\|\boldsymbol{\mu}\|_2 = K\sigma$ for some $0 < K \leq O(\sqrt{\log n})$. All of our results rely on a mild assumption which lower bounds the sparsity of the graph generated by the CSBM model. This assumption requires the expected degree of a node in the graph to be larger than $\log^2 n$ which covers reasonably sparse graphs. Note that we do not assume anything about the relative magnitude between p and q. All results hold regardless of $p \geq q$ or $p \leq q$.

Assumption 1. $p, q = \Omega(\log^2 n/n)$.

3.1 "Easy Regime"

In this regime $\big(\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})\big)$ we show that there exists a choice of attention architecture Ψ, which is able to separate the edges with high probability. The result is constructive - we show that a two-layer MLP attention is able to correctly classify all edges with high probability. In Section 4, we provide extensive experiments using the two-layer MLP attention architecture and empirically validate the theoretical claims.

Theorem 1. *Suppose that $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$. Then, there exists a choice of attention architecture Ψ such that with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim \mathsf{CSBM}(n, p, q, \boldsymbol{\mu}, \sigma^2)$ it holds that Ψ separates intra-class edges from inter-class edges.*

Proof sketch: To prove Theorem 1 we first transform the pair $(\mathbf{X}_i, \mathbf{X}_j)$ to a new pair $(\tilde{\boldsymbol{w}}^T\mathbf{X}_i, \tilde{\boldsymbol{w}}^T\mathbf{X}_j)$, where $\tilde{\boldsymbol{w}} = \text{sign}(p - q)\boldsymbol{\mu}/\|\boldsymbol{\mu}\|_2$ is a unit vector that maximizes the total pairwise distances among the four means given below.

[4] $\text{LeakyRelu}(x) = x$ if $x \geq 0$ and βx for some constant $\beta \in [0, 1)$ otherwise.

[5] If $p = q$, we simply require that $\text{sign}(\Psi(\mathbf{X}_i, \mathbf{X}_j)) = 1$ for one class of edges and $\text{sign}(\Psi(\mathbf{X}_i, \mathbf{X}_j)) = -1$ for the other class of edges. We define separability in this way because, as we will see later, it leads to desirable attention coefficients which in turn lead to perfect node classification result.

When we consider the pair space $\left(\tilde{\boldsymbol{w}}^T\mathbf{X}_i, \tilde{\boldsymbol{w}}^T\mathbf{X}_j\right)$, we can think of each pair as a two-dimensional Gaussian vector, whose means one of the following: $(\tilde{\boldsymbol{w}}^T\boldsymbol{\mu}, \tilde{\boldsymbol{w}}^T\boldsymbol{\mu})$, $(-\tilde{\boldsymbol{w}}^T\boldsymbol{\mu}, \tilde{\boldsymbol{w}}^T\boldsymbol{\mu})$, $(\tilde{\boldsymbol{w}}^T\boldsymbol{\mu}, -\tilde{\boldsymbol{w}}^T\boldsymbol{\mu})$, or $(-\tilde{\boldsymbol{w}}^T\boldsymbol{\mu}, -\tilde{\boldsymbol{w}}^T\boldsymbol{\mu})$. We need to classify the data corresponding to means $(\tilde{\boldsymbol{w}}^T\boldsymbol{\mu}, \tilde{\boldsymbol{w}}^T\boldsymbol{\mu})$ and $(-\tilde{\boldsymbol{w}}^T\boldsymbol{\mu}, -\tilde{\boldsymbol{w}}^T\boldsymbol{\mu})$ as $\operatorname{sign}(p-q)$ (i.e, intra-class edges) and classify the data corresponding to the other means as $\operatorname{sign}(q-p)$ (i.e., inter-edges). This problem is known in the literature as the *"XOR problem"* (Minsky & Papert, 1969). To achieve this we consider an architecture Ψ that separates the first and third quadrants (intra-edges) of the 2D space from the second and forth quadrants (inter-edges). The particular function Ψ has been chosen such that it is able to classify the means of the XOR problem correctly. At the same time, our assumption $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$ guarantees that the distance between the means of the XOR problem is much larger than the standard deviation of the Gaussians, thus, there is not much overlap between the distributions. This property guarantees that with high probability the sign of the expected $\Psi(\mathbf{X}_i, \mathbf{X}_j)$ is the same as the sign of Ψ applied to the means of $(\mathbf{X}_i, \mathbf{X}_j)$. Since the means are classified correctly, we use concentration arguments to prove that with high probability over $(\mathbf{X}, \mathbf{A}) \sim \mathsf{CSBM}(n, p, q, \boldsymbol{\mu}, \sigma^2)$, Ψ separates the edges.

We present two corollaries of the above theorem. The first corollary characterizes the attention coefficients induced by using the architecture Ψ from Theorem 1. In this regime, separability of the edges implies high concentration for the attention coefficients, γ_{ij}. This shows the desired behavior of the attention mechanism: when $p \geq q$, it maintains intra-class edges and essentially ignores all inter-class edges; when $p < q$, it maintains inter-class edges and essentially ignores all intra-class edges.

Corollary 2. *Suppose that $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$. Then there exists a choice of attention architecture Ψ such that with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim$ CSBM$(n, p, q, \boldsymbol{\mu}, \sigma^2)$ it holds that*

1. *If $p \geq q$, then $\gamma_{ij} = \frac{2}{np}(1 \pm o_n(1))$ if (i, j) is an intra-class edge and $\gamma_{ij} = o(\frac{1}{n(p+q)})$ otherwise;*
2. *If $p < q$, then $\gamma_{ij} = \frac{2}{nq}(1 \pm o_n(1))$ if (i, j) is an inter-class edge and $\gamma_{ij} = o(\frac{1}{n(p+q)})$ otherwise.*

Proof sketch: Corollary 2 is proved using the fact that $\Psi(\mathbf{X}_i, \mathbf{X}_j)$ concentrates around its expected value, which is close to, up to a small error, the function Ψ applied to the means of the data $(\mathbf{X}_i, \mathbf{X}_j)$. Given concentration of Ψ and concentration of node degrees that is guaranteed by Assumption 1, we can show concentration of γ_{ij} around the values reported in the corollary. If $p \geq q$, then for i, j in the same class, we have that $\Psi(\mathbf{X}_i, \mathbf{X}_j)$ concentrates around large positive value, which means that $\exp(\Psi(\mathbf{X}_i, \mathbf{X}_j)$ is exponentially large. On the other hand, by the definition of the attention coefficients (Equation 1), the denominator of γ_{ij} is dominated by terms (i, k) where k is in the same class as i (this is since for pairs i, k from different class $\exp(\Psi(\mathbf{X}_i, \mathbf{X}_k))$ is exponentially small when $p \geq q$), and since each node i is connected to $\Theta(np)$ many intra-class nodes and $\Psi(\mathbf{X}_i, \mathbf{X}_k)$ concentrates around the same value for each intra-class pair, we get the above value of γ_{ij}. A similar reasoning applies to inter-class pairs and yields the vanishing value of γ_{ij} for inter-class edges when $p \geq q$. The argument for $p < q$ follows in same way.

Note that, when $p \geq q$, the attention coefficients which correspond to intra-class edges are much larger than the attention coefficients of inter-class edges. This essentially means that the model ignores inter-class edges. Similarly, when $p < q$, the model ignores intra-class edges. By using Corollary 2 on γ_{ij} we obtain a *node classification* result as well. In the following, we say that the model *separates the nodes* if $\boldsymbol{h}'_i > 0$ when $i \in C_1$ and $\boldsymbol{h}'_i < 0$ when $i \in C_0$.

Corollary 3. *Suppose that $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$. Then, there exists a choice of attention architecture Ψ such that with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim$ CSBM$(n, p, q, \boldsymbol{\mu}, \sigma^2)$, the model separates the nodes for any p, q satisfying Assumption 1.*

The above result holds for any value of p, q satisfying Assumption 1. That is, even when the graph structure is noisy (i.e., have many inter-class edges) it is possible to obtain perfect node classification.

Proof sketch: The proof of this corollary is a consequence of Corollary 2. We discuss the case $p \geq q$ as the case $p < q$ follows similarly. Intuitively, Corollary 2 implies that the means of the convolved data should concentrate around the same means as the original data. On the other hand, by the choice of p, q we expect that each node will be connected to $\Theta(np)$ many intra-class nodes (and essentially no inter-class nodes, due to the small value of the attention coefficients for inter-class edges in Corollary 2, which implies the independence in q) so that the averaging operation reduces the variance significantly to $\approx \sigma^2/np$. However, since the distance between the new means is around $2\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$ and the variance is much smaller than σ^2, we can expect to achieve perfect node separability.

Nonetheless, in this regime, we prove that a simple linear classifier, which does not use the graph at all, achieves perfect node separability with high probability. In particular, by a classical argument (Anderson, 2003), the Bayes optimal classifier for the node features (*without the graph*) is realized by a simple linear classifier, which achieves perfect node separability with high probability.[6] This implies that in the above regime, using the graph model is unnecessary, as it does not provide additional power compared to a simple linear classifier for the node classification task.

Proposition 4. *Suppose $\|\boldsymbol{\mu}\|_2 = \Omega(\sigma\sqrt{\log n})$, and let $\mathbf{X}$ be sampled from the Gaussian mixture model. Then, the Bayes optimal classifier is realized by a linear classifier which achieves perfect node separability with probability at least $1 - o_n(1)$ over $\mathbf{X}$.*

3.2 "Hard Regime"

In this regime ($\|\boldsymbol{\mu}\|_2 = K\sigma$ for $K \leq O(\sqrt{\log n})$), we show that any attention architecture Ψ will fail to separate the edges. The next theorem quantifies the misclassification of edge pairs that Ψ exhibits. Below we define $\Phi_{\mathrm{c}}(\cdot) = 1 - \Phi(\cdot)$, where $\Phi(\cdot)$ denotes standard Gaussian CDF.

Theorem 5. *Suppose $\|\boldsymbol{\mu}\|_2 = K\sigma$ for some $K > 0$ and let Ψ be any attention mechanism. Then,*

1. *For any $c' > 0$, with probability at least $1 - O(n^{-c'})$, Ψ fails to correctly classify at least a $2 \cdot \Phi_{\mathrm{c}}(K)^2$ fraction of the inter-class edges.*
2. *For any $\kappa > 1$ if $q > \frac{\kappa \log^2 n}{n\Phi_{\mathrm{c}}(K)^2}$, then with probability at least $1 - O\left(\frac{1}{n^{\frac{\kappa}{4}\Phi_{\mathrm{c}}(K)^2 \log n}}\right)$, Ψ misclassify at least one inter-class edge.*

Part 1 of the theorem implies that if $\|\boldsymbol{\mu}\|_2$ is *linear* in the standard deviation σ, then with overwhelming probability the attention mechanism fails to distinguish a constant fraction of inter-edge pairs from the intra-edge pairs. Furthermore, part 2 of the theorem characterizes a regime for the inter-edge probability q where the attention mechanism fails to distinguish at least one inter-edge node pair, by providing a lower bound on q in terms of the scale at which the distance between the means grows compared to the standard deviation σ. This aligns with the intuition that as we increase the distance between the means, it gets easier for the attention mechanism to correctly distinguish the node pairs. However, if q is also increased in the right proportion (in other words, if the noise in the graph is increased), then the attention mechanism will still fail to correctly distinguish at least one of the inter-edge node pairs. For instance, setting $K = \sqrt{2\log\log n}$ and $\kappa = 4$, we get that if $q > \Omega\left(\frac{\log^{4+o(1)} n}{n}\right)$, then with probability at least $1/2$, Ψ misclassifies at least one inter-edge.

Proof sketch: Consider the pair of node features $(\mathbf{X}_i, \mathbf{X}_j)$. Recall that the goal of an attention mechanism is to distinguish the pairs where i and j are in the same class (intra-edges) from the pairs where i and j are in different classes (inter-edges). To show that every attention mechanism fails at this task, first, we use the underlying distribution of the data to characterize the Bayes optimal classifier for this problem and compute an upper bound on the probability with which the optimal classifier correctly classifies a single inter-edge pair. Then the proof of part 1 of the theorem follows from a concentration argument for the fraction of inter-edge pairs that are misclassified by the optimal classifier. For part 2, we use a similar concentration argument to choose a suitable threshold for q that forces the attention mechanism to fail on at least one inter-edge pair.

As a motivating example, we focus our attention on one of the most popular attention architecture (Velickovic et al., 2018), where α is a single layer neural network parametrized by $(\boldsymbol{w}, \boldsymbol{a}, b) \in \mathbb{R}^d \times \mathbb{R}^2 \times \mathbb{R}$ with LeakyRelu as activation. Namely, the attention coefficients are defined by

$$\gamma_{ij} \stackrel{\text{def}}{=} \frac{\exp\left(\mathrm{LeakyRelu}\left(\boldsymbol{a}^T \cdot \begin{bmatrix}\boldsymbol{w}^T\mathbf{X}_i \\ \boldsymbol{w}^T\mathbf{X}_j\end{bmatrix} + b\right)\right)}{\sum_{\ell \in N_i} \exp\left(\mathrm{LeakyRelu}\left(\boldsymbol{a}^T \cdot \begin{bmatrix}\boldsymbol{w}^T\mathbf{X}_i \\ \boldsymbol{w}^T\mathbf{X}_\ell\end{bmatrix} + b\right)\right)}. \tag{2}$$

We show that with a very high probability most of the attention coefficients γ_{ij} in Equation 2 are going to be $\Theta(1/|N_i|)$, which implies that the model fails to distinguish intra- and inter-edges.

Theorem 6 (informal)**.** *Assume that $\|\boldsymbol{\mu}\|_2 \leq K\sigma$ and $\sigma \leq K'$ for some constants K and K'. Moreover, assume that the parameters $(\boldsymbol{w}, \boldsymbol{a}, b)$ are bounded by a constant. Then, with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim$ CSBM$(n, p, q, \boldsymbol{\mu}, \sigma^2)$, at least 90% of γ_{ij} are $\Theta\left(1/|N_i|\right)$.*

[6] A Bayes optimal classifier makes the most probable prediction for a data point. Formally, for our scenario, such a classifier is given by $h^*(\boldsymbol{x}) = \arg\max_{c \in \{0,1\}} \mathbf{Pr}[y = c \mid \boldsymbol{x}]$.

Proof sketch: Theorem 6 is due to the inability of the attention mechanism to correctly classify intra- and inter-class edges as stated in Theorem 5. This means that the numerator in Equation 2 is not indicative of the class memberships of nodes i, j but rather acts like Gaussian noise. Combined with the observation that the denominator in Equation 2, which we will denote by δ_i, is the same for all γ_{il} where $l \in N_i$, this implies that most of the attention coefficients are roughly the same. Using concentration arguments about $\{\boldsymbol{w}^T \mathbf{X}_l\}_l$ yields $\gamma_{ij} = \Theta(1/\delta_i)$ and $\delta_i = \Theta(|N_i|)$.

Compared to the "easy regime", it is difficult to obtain a separation result for the nodes. In the "easy regime", the distance between the means was larger than the standard deviation, which made the "signal" (the expectation of the convolved data) dominate the "noise" (i.e., the variance of the convolved data). In the "hard regime" the "noise" dominates the "signal". Thus, we conjecture the following.

Conjecture 7. *Suppose that $\|\boldsymbol{\mu}\|_2 \leq K\sigma$ and $\sigma \leq K'$ for some constants K and K'. Then, any single layer graph attention model fails to perfectly classify the nodes with high probability when $|p - q| = O\left(\sigma\sqrt{(\log n)/\Delta}\right)$, where Δ is the expected degree.*

The above conjecture means implies that in the "hard regime" the performance of the graph attention model depends on q as opposed to the "easy regime", where in Theorem 3 we show that it doesn't. This property is verified by our synthetic experiments in Figures 1c and 1f. The $\sigma\sqrt{(\log n)/\Delta}$ bound comes from our conjecture that the expected maximum of the graph convolved data (with attention) over the nodes is at least $c\sigma\sqrt{(\log n)/\Delta}$ for some constant $c > 0$.

4 Experiments

In this section, we demonstrate empirically our results in Section 3 on synthetic and real data. The parameters of the models that we experiment with are set by using an ansatz based on our theorems. The particular details are given in Appendix B.1. We use the standard split which comes from PyTorch Geometric (Fey & Lenssen, 2019). Also, in Appendix B.3.2, we provide experiments on real data where PyTorch Geometric is used to train the models. The results are similar to the main paper, we provide discussion when there are discrepancies. With two exemptions in Figures 1e and 2b, in all our experiments we use MLP-GAT, where the attention mechanism Ψ is set to be a two-layer network using $\mathrm{LeakyRelu}$ activation function. The exemptions are made to demonstrate Theorem 6. We demonstrate more results for the original GAT (Velickovic et al., 2018) mechanism with two heads in Appendix B.2.

4.1 Synthetic data

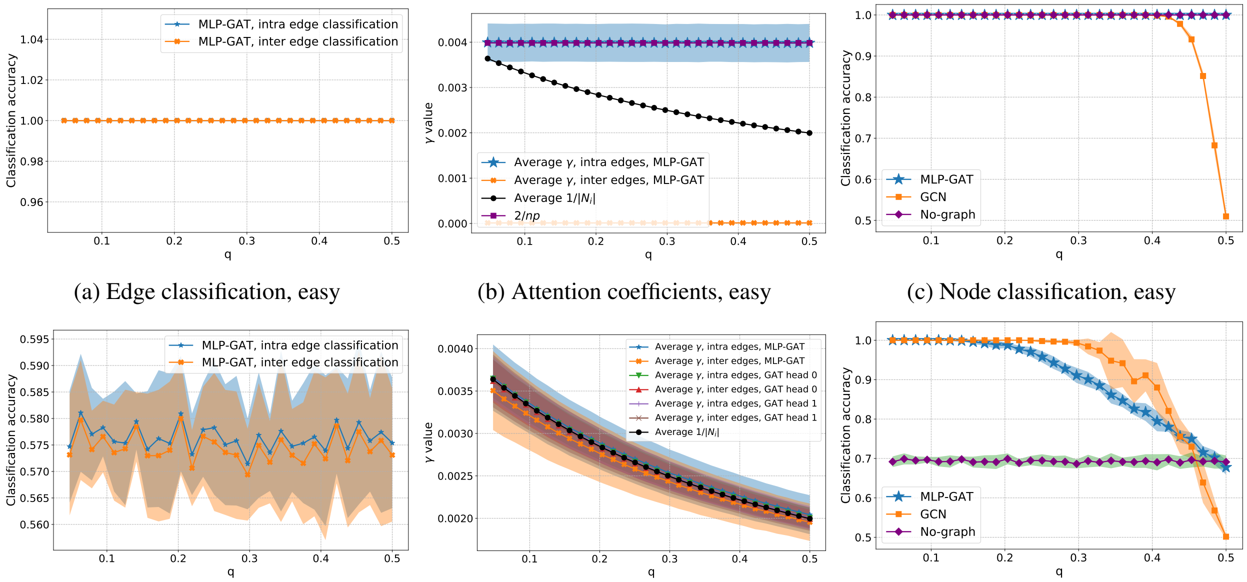
We use the CSBM to generate the data. We present two sets of experiments. In the first set we fix the distance between the means and vary q, and in the second set, we fix q and vary the distance. We set $n = 1000$, $d = n/\log^2(n)$, $p = 0.5$ and $\sigma = 0.1$. Results are averaged over 10 trials.

4.1.1 Fixing the distance between the means and varying q

We consider the two regimes separately, where for the "easy regime" we fix the mean $\boldsymbol{\mu}$ to be a vector where each coordinate is equal to $10\sigma\sqrt{\log n^2}/2\sqrt{d}$. This guarantees that the distance between the means is $10\sigma\sqrt{\log n^2}$. In the "hard regime" we fix the mean $\boldsymbol{\mu}$ to a vector where each coordinate is equal to $\sigma/\sqrt{d}$, and this guarantees that the distance is σ. We vary q from $\log^2(n)/n$ to p.

In Figure 1 we illustrate Theorem 1 and Corollaries 2, 3 for the "easy regime", and Theorems 5, 6 for the hard regime. In particular, in Figure 1a we show Theorem 1, MLP-GAT is able to classify intra and inter edges perfectly. In Figure 1b we show that in the "easy regime", the γ that correspond to intra-edges concentrate around $2/np$ for MLP-GAT, while the γ for the inter-edges concentrate to tiny values, as proved in Corollary 2. In Figure 1c we observe that the performance of MLP-GAT for node classification is independent of q in the "easy regime" as is proved in Corollary 3. However, in this plot, we observe that not using the graph also achieves perfect node classification, a result which is proved in Proposition 4. In the same plot, we also show the performance of uniform graph convolution (Kipf & Welling, 2017), where its performance depends on q (see Baranwal et al. (2021)). In Figure 1d we show Theorem 5, MLP-GAT misclassifies a constant fraction of the intra and inter edges as proved in Theorem 5. In Figure 1e we show Theorem 6, γ in the "hard regime" concentrate around uniform (GCN) coefficients for both MLP-GAT and GAT.

In Figure 1f we illustrate that node classification accuracy is a function of q for MLP-GAT. This is conjectured in Conjecture 7.



(a) Edge classification, easy

(b) Attention coefficients, easy

(c) Node classification, easy

(d) Edge classification, hard

(e) Attention coefficients, hard

(f) Node classification, hard

Figure 1: Demonstration of Theorem 1 and Corollaries 2, 3 for the "easy regime", and Theorems 5, 6 for the hard regime. The first row of figures demonstrates the former results and the second row of figures demonstrates the latter results. The shaded areas show standard deviation.

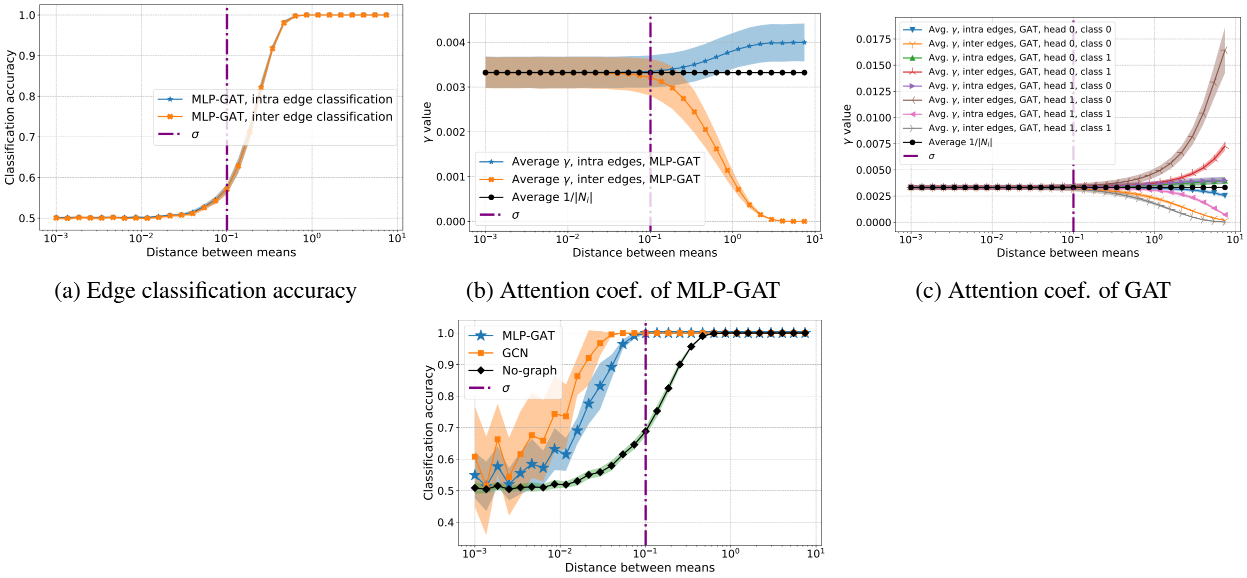
4.1.2 Fixing q and varying the distance between the means

We consider the case where $q = 0.1$. In Appendix B.2.2, we also demonstrate the case where $q = 0.4$. The results are similar in this case too. In Figure 2 we show how the attention coefficients of MLP-GAT and GAT, the node and edge classification depend on the distance between the means. We also add a vertical line at σ to approximately separate the easy (left of σ) and hard (right of σ) regimes. Figure 2a illustrates Theorems 1 and 5 in the "hard" and "easy" regimes, respectively. In particular, we observe that in the "hard regime" MLP-GAT fails to distinguish intra from inter edges, while in the "easy regime" it is able to do that perfectly for a large enough distance between the means.

In Figure 2b we observe that in the "hard regime" γ concentrate around the uniform (GCN) coefficients, while in the "easy regime" MLP-GAT is able to maintain the γ for the intra edges, while it sets the γ to tiny values for the inter edges. In Figure 2c. we observe that in the "hard regime" γ of GAT concentrate around the uniform coefficients (proved in Theorem 6), while in the "easy regime" although the γ concentrate, GAT is not able to distinguish intra from inter edges. This makes sense since the separation of edges can't be done by simple linear classifiers that GAT is using, see the discussion below Theorem 6. Finally, in Figure 2d we show node classification results for MLP-GAT. In the "easy regime" we observe perfect classification as proved in Corollary 3. However, as the distance between the means decreases, we observe that MLP-GAT starts to misclassify nodes.

4.2 Real data

In this experiment, we illustrate the attention coefficients, node and edge classification for MLP-GAT as a function of the distance between the means on real data. We use the popular real data Cora, PubMed, and CiteSeer. These data are publicly available and can be downloaded from Fey & Lenssen (2019). The datasets come with multiple classes, however, for each of our experiments we do a one-v.s.-all classification for a single class. This is a semi-supervised problem, only a fraction of the training nodes have labels. The rest of the nodes are used for measuring prediction accuracy. To control the distance between the means of problem we use the true labels to determine the class of each node and then we compute the empirical mean for each class. We subtract the empirical means from their



(a) Edge classification accuracy

(b) Attention coef. of MLP-GAT

(c) Attention coef. of GAT

(d) Node classification accuracy

Figure 2: Attention coefficients of MLP-GAT and GAT, node and edge classification as a function of the distance between the means. The shaded areas in our plots show standard deviation.

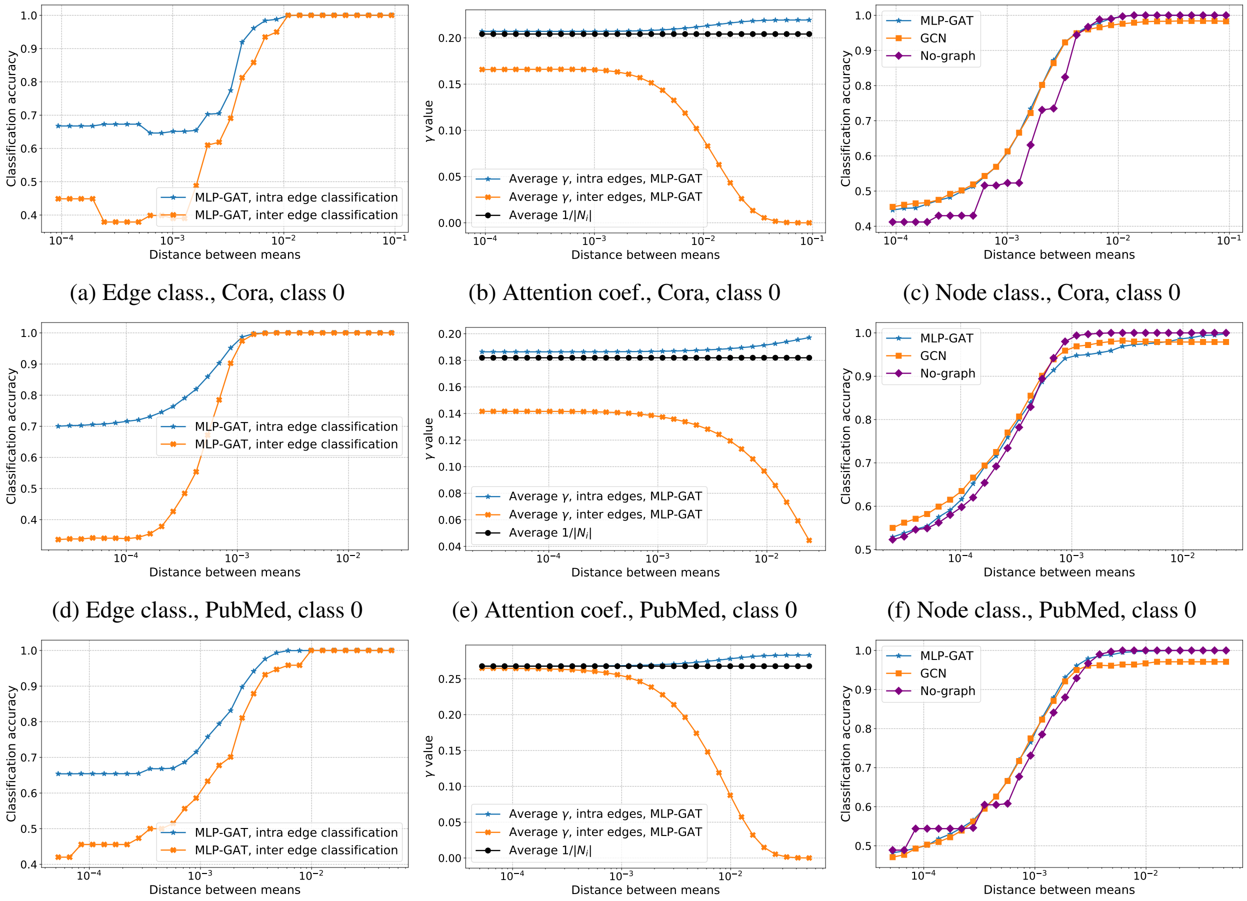
corresponding classes and we also add means $\boldsymbol{\mu}$ and $-\boldsymbol{\mu}$ to each class, respectively. This modification can be thought of as translating the mean of the distribution of the data for each class.

The results of this experiment are shown in Figure 3. In this figure we show results only for class 0 of each dataset, for results on other classes see Appendix B.3. The results are similar. We note that in the real data we also observe similar behavior of MLP-GAT in the easy and hard regimes as for the synthetic data. In particular, for all datasets as the distance of means increases, MLP-GAT is able to accurately classify the intra and inter edges, see Figures 3a, 3d and 3g. Moreover, as the distance between the means increases, the average intra γ becomes much larger than the average inter γ, see Figures 3b, 3e and 3h, and the model is able to classify the nodes accurately, see Figures 3c, 3f and 3i. On the contrary, in the same figures, we observe that as the distance of the means decreases then MLP-GAT is not able to separate intra from inter edges, the averaged γ are very close to uniform coefficients and the model can't classify the nodes accurately.

Note that Figure 3 does not show the standard deviation for the attention coefficients γ. We show the standard deviation of γ in Figure 4. We observe that the standard deviation is higher than what we observed in the synthetic data. In particular, it can be more than half of the averaged γ. This is to be expected since for the real data the degrees of the nodes do not concentrate as well. In Figure 4 we show that the standard deviation of the uniform coefficients $1/|N_i|$ is also high and that the standard deviation of γ is similar to that of $1/|N_i|$ for intra-class γ, while the deviation for inter-class γ is large for a small distance between the means, but it gets much smaller as the distance increases.

5 Conclusion and future work

We show that graph attention improves robustness to noise in graph structure in an "easy" regime, where the graph is not needed at all. We also show that graph attention may not be very useful in a "hard" regime where the node features are noisy. Our work shows that single-layer graph attention has limited power at distinguishing intra- from inter-class edges. Given the empirical successes of graph attention and its many variants, a promising future work is to study the power of multi-layer graph attention mechanisms for distinguishing intra- and inter-class edges.



(a) Edge class., Cora, class 0

(b) Attention coef., Cora, class 0

(c) Node class., Cora, class 0

(d) Edge class., PubMed, class 0

(e) Attention coef., PubMed, class 0

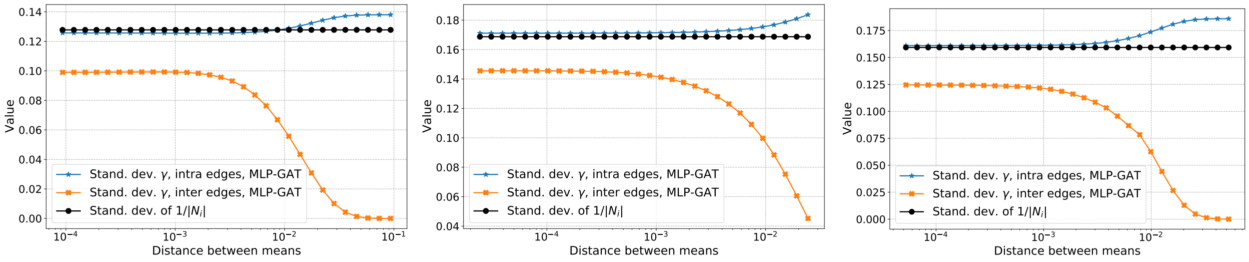
(f) Node class., PubMed, class 0

(g) Edge class., CiteSeer, class 0

(h) Attention coef., CiteSeer, class 0

(i) Node class., CiteSeer, class 0

Figure 3: Attention coefficients, node and edge classification for MLP-GAT as a function of the distance between the means for real data.



(a) Cora

(b) PubMed

(c) CiteSeer

Figure 4: Standard deviation for attention coefficients of MLP-GAT.

REFERENCES

E. Abbe. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18:1–86, 2018.

N. Alon and J. H. Spencer. *The Probabilistic Method.* John Wiley & Sons, 2004.

T.W. Anderson. *An introduction to multivariate statistical analysis*. John Wiley & Sons, 2003.

A. Athreya, D. E. Fishkind, M. Tang, C. E. Priebe, Y. Park, J. T. Vogelstein, K. Levin, V. Lyzinski, Y. Qin, and D. L. Sussman. Statistical inference on random dot product graphs: A survey. *Journal of Machine Learning Research*, 18(226):1–92, 2018.

J. Atwood and D. Towsley. Diffusion-convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2001–2009, 2016.

D. Bahdanau, K. H. Cho, , and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations (ICLR)*, 2015.

A. Baranwal, K. Fountoulakis, and A. Jagannath. Graph convolution for semi-supervised classification: Improved linear separability and out-of-distribution generalization. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, pp. 684–693, 2021.

N. Binkiewicz, J. T. Vogelstein, and K. Rohe. Covariate-assisted spectral clustering. *Biometrika*, 104:361–377, 2017.

X. Bresson and T. Laurent. Residual gated graph convnets. In *arXiv:1711.07553*, 2018.

S. Brody, U. Alon, and E. Yahav. How attentive are graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2022.

J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations (ICLR)*, 2014.

Z. Chen, L. Li, and J. Bruna. Supervised community detection with line graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2019.

E. Chien, J. Peng, P. Li, and O. Milenkovic. Adaptive universal generalized pagerank graph neural network. In *International Conference on Learning Representations (ICLR)*, 2021.

M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3844–3852, 2016.

Y. Deshpande, A. Montanari S. Sen, and E. Mossel. Contextual stochastic block models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 45, pp. 2224–2232, 2015.

M. Fey and J. E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

V. Garg, S. Jegelka, and T. Jaakkola. Generalization and representational limits of graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 119, pp. 3419–3430, 2020.

M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In *IEEE International Joint Conference on Neural Networks (IJCNN)*, 2005.

W. L. Hamilton, R. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1025–1035, 2017.

R. Van Handel. Probability in high dimension. Technical report, PRINCETON UNIV NJ, 2014.

M. Henaff, J. Bruna, and Y. LeCun. Deep convolutional networks on graph-structured data. In *arXiv:1506.05163*, 2015.

Y. Hou, J. Zhang, J. Cheng, K. Ma, R. T. B. Ma, H. Chen, and M.-C. Yang. Measuring and improving the use of graph information in graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2019.

W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, and J. Leskovec. Open graph benchmark: datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

B. Knyazev, G. W. Taylor, and M. Amer. Understanding attention and generalization in graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 4202–4212, 2019.

B. J. Lee, R. A. Rossi, S. Kim, K. N. Ahmed, and E. Koh. Attention models in graphs: A survey. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2019.

Y. Li, R. Zemel, M. Brockschmidt, and D. Tarlow. Gated graph sequence neural networks. In *International Conference on Learning Representations (ICLR)*, 2016.

A. Loukas. How hard is to distinguish graphs with graph neural networks? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020a.

A. Loukas. What graph neural networks cannot learn: Depth vs width. In *International Conference on Learning Representations (ICLR)*, 2020b.

M. Minsky and S. Papert. Perceptron: an introduction to computational geometry, 1969.

C. Moore. The computer science and physics of community detection: Landscapes, phase transitions, and hardness. *Bulletin of The European Association for Theoretical Computer Science*, 1(121), 2017.

O. Puny, H. Ben-Hamu, and Y. Lipman. Global attention improves graph networks generalization. In *arXiv:2006.07846*, 2020.

P. Rigollet and J.-C. Hütter. High dimensional statistics. *Lecture notes for course 18S997*, 813:814, 2015.

F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1), 2009.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 6000–6010, 2017.

P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

R. Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

M. Wang, D. Zheng, Z. Ye, Q. Gan, M. Li, X. Song, J. Zhou, C. Ma, L. Yu, Y. Gai, T. Xiao, T. He, G. Karypis, J. Li, and Z. Zhang. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*, 2019.

K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

J. Zhu, Y. Yan, L. Zhao, M. Heimann, L. Akoglu, and D. Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

A Proofs

A.1 General Results

We start by stating some standard definitions and probability tools which will be used throughout. The first definition is regarding *sub-Gaussian* random variables. These random variables are characterized by their tail decay.

Definition A.1 (Vershynin (2018)). *We say that a random variable $\boldsymbol{z}$ follows a* sub-Gaussian *distribution if there are positive constants C, v such that for every $t > 0$*

$$\mathbf{Pr}[|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| > t] \leq C \exp(-vt^2).$$

Equivalently, $\boldsymbol{z}$ is sub-Gaussian if $\mathbf{E}[\exp(a(\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}])^2)] \leq 2$ for some $a > 0$. This condition is known as ψ_2-condition.

The following lemma discuss the behavior of the maxima of sub-Gaussian random variables.

Lemma A.2 (Rigollet & Hütter (2015)). *Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ be sub-Gaussian random variables with the same mean and sub-Gaussian parameter $\tilde{\sigma}^2$. Then,*

$$\mathbf{E}\left[\max_{i \in [n]} (\boldsymbol{x}_i - \mathbf{E}[\boldsymbol{x}_i])\right] \leq \tilde{\sigma}\sqrt{2 \log n}.$$

Moreover, for any $t > 0$

$$\mathbf{Pr}\left[\max_{i \in [n]} (\boldsymbol{x}_i - \mathbf{E}[\boldsymbol{x}_i]) > t\right] \leq 2n \exp\left(-\frac{t^2}{2\tilde{\sigma}^2}\right).$$

Next, we define *Lipschitz* functions and state the LeakyRelu is Lipschitz.

Definition A.3. *Let $(\mathcal{X}, d_{\mathcal{X}})$ and $(\mathcal{Y}, d_{\mathcal{Y}})$ be metric spaces. A function $f : \mathcal{X} \to \mathcal{Y}$ is called L-*Lipschitz *if there is $L \in \mathbb{R}$ such that for every $u, v \in \mathcal{X}$*

$$d_{\mathcal{Y}}(f(u), f(v)) \leq L \cdot d_{\mathcal{X}}(u, v).$$

Fact A.4. *LeakyRelu is L-Lipschitz with $L \leq 1$.*

Next, for completeness, we state two forms (additive and multiplicative) of Chernoff bounds used in this work.

Lemma A.5 (Alon & Spencer (2004); Handel (2014); Vershynin (2018)). *Let $\chi_1, \ldots, \chi_n$ be identical independent random variables ranging in $[0, 1]$, and let $p = \mathbf{E}[\chi_1]$. Then for any $\epsilon \in (0, 1)$, it holds that*

$$\mathbf{Pr}\left[\left|\frac{1}{n}\sum_{i \in [n]} \chi_i - p\right| > \epsilon\right] < 2 \exp\left(-\frac{\epsilon^2 n}{4}\right),$$

and for any $\gamma \in (0, 2]$, it holds that

$$\mathbf{Pr}\left[\left|\frac{1}{n}\sum_{i \in [n]} \chi_i - p\right| > \gamma p\right] < 2 \exp\left(-\frac{\gamma^2 p n}{4}\right).$$

In order to prove Theorem 1 we will need the following concentration result on LeakyRelu whose constant denoted by β. Fix $(\boldsymbol{w}, \boldsymbol{a}) \in \mathbb{R}^d \times \mathbb{R}^2$ and for $i, j \in [n]$ let

$$\boldsymbol{z}_{ij} = \boldsymbol{a}_1 \boldsymbol{w}^T \mathbf{X}_i + \boldsymbol{a}_2 \boldsymbol{w}^T \mathbf{X}_j \sim \begin{cases} N((\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu},\ \sigma^2 \|\boldsymbol{a}\|^2 \|\boldsymbol{w}\|^2) & \text{if } i, j \in C_1 \\ N((\boldsymbol{a}_1 - \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu},\ \sigma^2 \|\boldsymbol{a}\|^2 \|\boldsymbol{w}\|^2) & \text{if } i \in C_1,\ j \in C_0 \\ N(-(\boldsymbol{a}_1 - \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu},\ \sigma^2 \|\boldsymbol{a}\|^2 \|\boldsymbol{w}\|^2) & \text{if } i \in C_0,\ j \in C_1 \\ N(-(\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu},\ \sigma^2 \|\boldsymbol{a}\|^2 \|\boldsymbol{w}\|^2) & \text{if } i, j \in C_0 \end{cases}.$$

Lemma A.6. *There exists an absolute constant $C > 0$ such that with probability at least $1 - o_n(1)$, we have*

$$\begin{aligned}
\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) &= \mathrm{LeakyRelu}\left((\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu}\right) \pm C\sigma\|\boldsymbol{a}\|\|\boldsymbol{w}\|\sqrt{2\log n}, \quad \textit{if } i, j \in C_1, \\
\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) &= \mathrm{LeakyRelu}\left((\boldsymbol{a}_1 - \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu}\right) \pm C\sigma\|\boldsymbol{a}\|\|\boldsymbol{w}\|\sqrt{2\log n}, \quad \textit{if } i \in C_1, j \in C_0, \\
\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) &= \mathrm{LeakyRelu}\left(-(\boldsymbol{a}_1 - \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu}\right) \pm C\sigma\|\boldsymbol{a}\|\|\boldsymbol{w}\|\sqrt{2\log n}, \quad \textit{if } i \in C_0, j \in C_1, \\
\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) &= \mathrm{LeakyRelu}\left(-(\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T \boldsymbol{\mu}\right) \pm C\sigma\|\boldsymbol{a}\|\|\boldsymbol{w}\|\sqrt{2\log n}, \quad \textit{if } i, j \in C_0.
\end{aligned}$$

Proof: Since for every $i, j \in [n]^2$ the random variable $\boldsymbol{z}_{ij}$ follows a normal distribution, by definition it is sub-Gaussian with parameter $c \cdot \sqrt{\mathbf{Var}[\boldsymbol{z}_{ij}]}$ for $c > 1$ large enough constant (see definition A.1). By Fact A.4, LeakyRelu is L-Lipschitz function with $L \leq 1$

$$\begin{aligned}\mathop{\mathbf{E}}_{\boldsymbol{z}}\left[\exp\left(\frac{(\mathrm{LeakyRelu}(\boldsymbol{z}) - \mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z})])^2}{K^2}\right)\right] &= \mathop{\mathbf{E}}_{\boldsymbol{z}}\left[\exp\left(\frac{\mathbf{E}_{\boldsymbol{z}'}[\mathrm{LeakyRelu}(\boldsymbol{z}) - \mathrm{LeakyRelu}(\boldsymbol{z}')]^2}{K^2}\right)\right] \\ &= \mathop{\mathbf{E}}_{\boldsymbol{z}}\left[\exp\left(\frac{L^2(\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}])^2}{K^2}\right)\right]. \end{aligned} \tag{3}$$

Setting $K = c \cdot \sqrt{\mathbf{Var}[\boldsymbol{z}]} \cdot L$, implies that (3) is bounded above by 2, which means that Leaky-Relu is sub-Gaussian with parameter $c \cdot \sqrt{\mathbf{Var}[\boldsymbol{z}]} \cdot L$ (see Vershynin (2018)). Therefore for any $t > 0$,

$$\mathop{\mathbf{Pr}}_{\boldsymbol{z}}\left[|\mathrm{LeakyRelu}(\boldsymbol{z}) - \mathbf{E}\left[\mathrm{LeakyRelu}(\boldsymbol{z})\right]| \geq t\right] \leq 2\exp\left(-\frac{t^2}{c^2 L^2\, \mathbf{Var}[\boldsymbol{z}]}\right). \tag{4}$$

Setting $t = 10cL\sqrt{\mathbf{Var}[\boldsymbol{z}] \log n}$, and applying a union bound over all $i, j \in [n]^2$, we get that with probability at least $1 - \frac{2}{n^{98}}$, the complement of (4) holds for all $i, j \in [n]^2$.

Next, we estimate $\mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z})]$. For any $t' > 0$ we have

$$\mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z})] = \mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z}) \cdot \mathbf{1}_{\{|\boldsymbol{z} - \mathbb{E}[\boldsymbol{z}]| \leq t'\}}] + \mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z}) \cdot \mathbf{1}_{\{|\boldsymbol{z} - \mathbb{E}[\boldsymbol{z}]| > t'\}}].$$

We consider both terms separately. First, note

$$\mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z}) \cdot \mathbf{1}_{\{|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| \leq t'\}}] = \mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z}) \mid |\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| \leq t'] \cdot \mathbf{Pr}[|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| \leq t'].$$

By writing $\boldsymbol{z} = \mathbf{E}[\boldsymbol{z}] + \sqrt{\mathbf{Var}[\boldsymbol{z}]} \cdot g$, for $g \sim N(0,1)$ and using Lipschitz continuity of the Leaky-Relu

$$\begin{aligned}\mathbf{E}\left[\mathrm{LeakyRelu}(\boldsymbol{z}) \mid |\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| \leq t'\right] &= \mathbf{E}\left[\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}] + \sqrt{\mathbf{Var}[\boldsymbol{z}]} \cdot g) \mid \sqrt{\mathbf{Var}[\boldsymbol{z}]}|g| \leq t'\right] \\ &\in \left[\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}]) - Lt',\ \mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}]) + Lt'\right]. \end{aligned} \tag{5}$$

Hence by using sub-Gaussian concentration,

$$\begin{aligned}\mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z}) \cdot \mathbf{1}_{\{|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| \leq t'\}}] &\geq \left(1 - 2e^{\left(-\frac{t'^2}{2\,\mathbf{Var}[\boldsymbol{z}]}\right)}\right)\left(\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}] - Lt')\right), \\ \mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z}) \cdot \mathbf{1}_{\{|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| \leq t'\}}] &\leq \mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}]) + Lt'. \end{aligned} \tag{6}$$

For the second summand, we apply Cauchy-Schwartz inequality and Lipshitzness of LeakyRelu

$$\begin{aligned}\left|\mathbf{E}\left[\mathrm{LeakyRelu}(\boldsymbol{z}) \cdot \mathbf{1}_{\{|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| > t'\}}\right]\right| &\leq \sqrt{\mathbf{E}[|\mathrm{LeakyRelu}(\boldsymbol{z})|^2] \cdot \mathbf{Pr}[|\boldsymbol{z} - \mathbf{E}[\boldsymbol{z}]| > t']} \\ &\leq \sqrt{2L^2\, \mathbf{E}[|\boldsymbol{z}|^2] \exp\left(-\frac{t'^2}{2\,\mathbf{Var}[\boldsymbol{z}]}\right)} \\ &\leq \sqrt{2L^2(\mathbf{E}[\boldsymbol{z}]^2 + \mathbf{Var}[\boldsymbol{z}]) \exp\left(-\frac{t'^2}{2\,\mathbf{Var}[\boldsymbol{z}]}\right)} \\ &\leq L\,\mathbf{E}[\boldsymbol{z}]\sqrt{2\exp\left(-\frac{t'^2}{2\,\mathbf{Var}[\boldsymbol{z}]}\right)} + L\sqrt{2\,\mathbf{Var}[\boldsymbol{z}] \exp\left(-\frac{t'^2}{2\,\mathbf{Var}[\boldsymbol{z}]}\right)}. \end{aligned} \tag{7}$$

Setting $t' = 10\sqrt{2\,\mathbf{Var}[\boldsymbol{z}] \log n}$, and combining Equations (6) and (7) results in

$$\mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z})] \leq \mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}]) + 10L\sqrt{2\,\mathbf{Var}[\boldsymbol{z}] \log n} + \frac{L\sqrt{2}\,\mathbf{E}[\boldsymbol{z}]}{n^{50}} + \frac{L\sqrt{2\,\mathbf{Var}[\boldsymbol{z}]}}{n^{50}}, \tag{8}$$

$$\mathbf{E}[\mathrm{LeakyRelu}(\boldsymbol{z})] \geq \left(1 - \frac{2}{n^{100}}\right)\left(\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}]) - 10L\sqrt{2\,\mathbf{Var}[\boldsymbol{z}] \log n}\right) - \frac{L\sqrt{2}(\mathbf{E}[\boldsymbol{z}] + \sqrt{\mathbf{Var}[\boldsymbol{z}]})}{n^{50}}. \tag{9}$$

Combining Equations (5), (8), (9) using the choice of t, we have that with a probability of at least $1 - O(1/n^{98})$ for all $i, j \in [n]$

$$\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) \leq \mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}_{ij}]) + 20L(c+1)\sqrt{2\,\mathbf{Var}[\boldsymbol{z}_{ij}]\log n} + \frac{L\sqrt{2}\left(\mathbf{E}[\boldsymbol{z}_{ij}] + \sqrt{\mathbf{Var}[\boldsymbol{z}_{ij}]}\right)}{n^{50}}, \tag{10}$$

$$\begin{aligned}\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) \geq \left(1 - \frac{2}{n^{100}}\right)\left(\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}_{ij}]) - 20(c+1)L\sqrt{2\,\mathbf{Var}[\boldsymbol{z}_{ij}]\log n}\right)\\ - \frac{L\sqrt{2}(\mathbf{E}[\boldsymbol{z}_{ij}] + \sqrt{\mathbf{Var}[\boldsymbol{z}_{ij}]})}{n^{50}}.\end{aligned} \tag{11}$$

We henceforth condition on this event. Recall that we have that

$$\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}_{ij}]) = \mathrm{LeakyRelu}((\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T\boldsymbol{\mu}) \qquad \text{for } i, j \in C_1 \tag{12}$$

$$\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}_{ij}]) = \mathrm{LeakyRelu}((\boldsymbol{a}_1 - \boldsymbol{a}_2)\boldsymbol{w}^T\boldsymbol{\mu}) \qquad \text{for } i \in C_1,\ j \in C_0 \tag{13}$$

$$\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}_{ij}]) = \mathrm{LeakyRelu}(-(\boldsymbol{a}_1 - \boldsymbol{a}_2)\boldsymbol{w}^T\boldsymbol{\mu}) \qquad \text{for } i \in C_0,\ j \in C_1 \tag{14}$$

$$\mathrm{LeakyRelu}(\mathbf{E}[\boldsymbol{z}_{ij}]) = \mathrm{LeakyRelu}(-(\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T\boldsymbol{\mu}) \qquad \text{for } i, j \in C_0. \tag{15}$$

Using Equations (10)-(15) we have that for $i, j \in C_1$

$$\mathrm{LeakyRelu}(\boldsymbol{z}_{ij}) = \mathrm{LeakyRelu}\left((\boldsymbol{a}_1 + \boldsymbol{a}_2)\boldsymbol{w}^T\boldsymbol{\mu}\right) \pm 20L(c+1)\sigma\|\boldsymbol{a}\|\|\boldsymbol{w}\|\sqrt{2\log n}.$$

The results for all other cases of i, j follow similarly. ■

The following statement considers the optimal Bayes classifier for data generated by the Gaussian mixture model.

Lemma A.7 (See section 6.4 in Anderson (2003))**.** *Let* $(\mathbf{X}, \mathbf{A}) \sim \mathit{CSBM}(n, p, q, \boldsymbol{\mu}, \sigma^2)$*. Then, the optimal Bayes classifier for* $\mathbf{X}$ *is realized by the linear classifier.*

$$h(\boldsymbol{x}) = \begin{cases} 0 & \textit{if } \boldsymbol{x}^T\boldsymbol{\mu} \leq 0 \\ 1 & \textit{if } \boldsymbol{x}^T\boldsymbol{\mu} > 0 \end{cases}.$$

Proof: For a given data point $\boldsymbol{x}$ and label $y \in \{0, 1\}$, the Bayes classifier is given by

$$h\left(\boldsymbol{x}\right) = \underset{c \in \{0,1\}}{\arg\max}\, \mathbf{Pr}\left[y = c \mid \boldsymbol{x}\right].$$

Note that since the class membership variables $\epsilon_1, \ldots, \epsilon_n \sim \mathrm{Ber}(1/2)$ are independent, we have $\mathbf{Pr}[y = 0] = \frac{1}{2}$ and $\mathbf{Pr}[y = 1] = \frac{1}{2}$. Therefore, by Bayes rule

$$\mathbf{Pr}[y = c \mid \boldsymbol{x}] = \frac{\mathbf{Pr}[y = c] \cdot f_{\boldsymbol{x}|y}(\boldsymbol{x} \mid y = c)}{\mathbf{Pr}[y = 0] f_{\boldsymbol{x}|y=0}(\boldsymbol{x} \mid y = 0) + \mathbf{Pr}[y = 1] f_{\boldsymbol{x}|y=1}(\boldsymbol{x} \mid y = 1)} = \frac{1}{1 + \frac{f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=1-c)}{f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=c)}}.$$

Assume that $\boldsymbol{x} \in C_0$, we have that $h(\boldsymbol{x}) = 0$ if and only if $\mathbf{Pr}[y = 0 \mid \boldsymbol{x}] \geq 1/2$. Therefore, if we consider class $c = 0$ we need that $\frac{f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=1)}{f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=0)} \leq 1$. That is,

$$\frac{f_{\boldsymbol{x}|y}(\boldsymbol{x} \mid y = 1)}{f_{\boldsymbol{x}|y}(\boldsymbol{x} \mid y = 0)} = \frac{\exp\left(-\frac{1}{2\sigma^2}\|\boldsymbol{x} - \boldsymbol{\mu}\|_2^2\right)}{\exp\left(-\frac{1}{2\sigma^2}\|\boldsymbol{x} + \boldsymbol{\mu}\|_2^2\right)} = \exp\left(-\frac{1}{2\sigma^2}\left(\|\boldsymbol{x} - \boldsymbol{\mu}\|_2^2 - \|\boldsymbol{x} + \boldsymbol{\mu}\|_2^2\right)\right) \leq 1,$$

which implies that $\boldsymbol{x}^T\boldsymbol{\mu} \leq 0$. Similarly, for label $c = 1$ we get that $\boldsymbol{x}^T\boldsymbol{\mu} > 0$. Hence, the Bayes classifier is given by

$$h(\boldsymbol{x}) = \begin{cases} 0 & \text{if } \boldsymbol{x}^T\boldsymbol{\mu} \leq 0 \\ 1 & \text{if } \boldsymbol{x}^T\boldsymbol{\mu} > 0 \end{cases},$$

which is a linear classifier. ■

The next lemma relates the fraction of misclassifications of the Bayes optimal classifier to the norm $\|\boldsymbol{\mu}\|_2$ (and thus to the distance between the means).

Lemma A.8. *Assuming independence of the underlying data* $\mathbf{X}$*, the following holds for the Bayes classifier.*

1. *If* $\|\boldsymbol{\mu}\|_2 \geq \sigma\sqrt{2\log n}$ *then with a probability of at least* $1 - o_n(1)$*, the Bayes classifier separates the data.*
2. *If* $\|\boldsymbol{\mu}\|_2 = K\sigma$ *for* $\omega(1) < K < \sigma\sqrt{2\log n}$*, then for any* $\kappa > 1$ *with a probability of at least* $1 - O\left(\frac{1}{n^{\kappa\Phi'/4}}\right)$ *the number of misclassified nodes is* $\Phi' n \left(1 \pm \sqrt{\frac{4\kappa \log n}{\Phi' n}}\right)$*, where* $\Phi' \overset{\text{def}}{=} 1 - \Phi(K)$ *and* Φ *denote the CDF of standard Gaussian.*
3. *If* $\|\boldsymbol{\mu}\|_2 = K\sigma$ *for* $K = O(1)$*, then with a probability of at least* $1 - o_n(1)$*, the number of misclassified nodes is at least* $\Phi' n(1 - o_n(1))$ *where* $\Phi' \geq \left(\frac{K}{K^2+1}\right) \cdot \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{K^2}{2}\right)$*.*

Proof: Fix $i \in [n]$ and write

$$\boldsymbol{x}_i = (2\epsilon_i - 1)\boldsymbol{\mu} + \sigma \boldsymbol{g}_i \qquad \text{where} \qquad \boldsymbol{g}_i \sim N(0, \mathbf{I}).$$

Assume $\epsilon_i = 0$ and consider the Bayes classifier from Lemma A.7. Then, the probability of misclassification is

$$\mathbf{Pr}[\boldsymbol{x}_i^T \boldsymbol{\mu} > 0] = \mathbf{Pr}\left[\frac{\boldsymbol{g}_i^T \boldsymbol{\mu}}{\|\boldsymbol{\mu}\|_2} > \frac{\|\boldsymbol{\mu}\|_2}{\sigma}\right] = 1 - \Phi\left(\frac{\|\boldsymbol{\mu}\|_2}{\sigma}\right),$$

where the last equality follows from the fact that $\frac{\boldsymbol{g}_i^T \boldsymbol{\mu}}{\|\boldsymbol{\mu}\|_2} \sim N(0, 1)$.

Suppose $\|\boldsymbol{\mu}\|_2 \geq \sigma\sqrt{2\log n}$. By using standard tail bounds for normal distribution (Vershynin, 2018),

$$1 - \Phi\left(\frac{\|\boldsymbol{\mu}\|_2}{\sigma}\right) \leq \frac{\sigma}{\|\boldsymbol{\mu}\|_2 \cdot \sqrt{2\pi}} \exp\left(-\frac{\|\boldsymbol{\mu}\|_2^2}{2\sigma^2}\right) \leq \frac{n^{-1}}{\sqrt{4\pi \log n}}.$$

Therefore, the probability that there exists $i \in C_0$ which is misclassified is at most $\frac{1}{2\sqrt{4\pi \log n}} = o(1)$. A similar argument can be applied to the case where $i \in C_1$, and an application of a union bound on the events that there is $i \in [n]$ which is misclassified finishes the proof of this case.

Next, consider the case where $\|\boldsymbol{\mu}\|_2 = K\sigma$ for $\omega(1) < K < \sigma\sqrt{2\log n}$. We have that for class $\epsilon_i = 0$ the misclassification probability is

$$\Phi' \overset{\text{def}}{=} 1 - \Phi(K).$$

Therefore, by applying additive Chernoff bound, we have that for any $\kappa > 1$

$$\mathbf{Pr}\left[\sum_{i \in C_0} \mathbf{1}_{i \text{ misclassified}} \notin \left(\frac{\Phi' n(1 \pm o(1))}{2} \pm \sqrt{\kappa \Phi' n \log n}\right)\right] \leq \frac{2}{n^{\kappa\Phi'/4}},$$

and similarly for $\epsilon_i = 1$. Applying a union bound over the two classes finishes the proof of this case.

Finally, consider the case where $\|\boldsymbol{\mu}\|_2 = K\sigma$ for some constant $K > 0$. For class $\epsilon_i = 0$, we have that the misclassification probability is lower-bounded by

$$\Phi' \overset{\text{def}}{=} 1 - \Phi(K) \geq \left(\frac{K}{K^2+1}\right) \cdot \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{K^2}{2}\right) = \Omega(1).$$

Therefore, by applying the Chernoff bound, we have that with a probability of at least $1 - o(1)$ we have that

$$\mathbf{Pr}\left[\sum_{i \in C_0} \mathbf{1}_{i \text{ misclassified}} < \frac{\Phi' n}{2}(1 - o(1))\right] = 1/\text{poly}(n).$$

By a similar argument for $\epsilon_i = 1$ and a union bound, the result follows. ■

The next observation will be useful throughout the paper.

Observation A.9. *Fix* $\boldsymbol{w} \neq 0$ *in* $\mathbb{R}^d$*, and let* $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_n$ *be iid drawn from* $N(0, \mathbf{I}_d)$*. Then* $\{\boldsymbol{w}^T \boldsymbol{g}_i\}$ *are independent.*

Proof: Note that for each $i \in [n]$ the random variable $\boldsymbol{w}^T \boldsymbol{g}_i$ follows Gaussian distribution with mean 0 and variance $\|\boldsymbol{w}\|_2$, it suffices to show that the covariance $\mathbf{E}[\boldsymbol{w}^T \boldsymbol{g}_i \cdot \boldsymbol{w}^T \boldsymbol{g}_j] = 0$ for all $i \neq j$. By definition

$$\mathbf{E}[\boldsymbol{w}^T \boldsymbol{g}_i \cdot \boldsymbol{w}^T \boldsymbol{g}_j] = \mathbf{E}\left[\sum_{k\in[d]}\sum_{\ell\in[d]} \boldsymbol{w}_k \boldsymbol{w}_\ell \boldsymbol{g}_{ik} \boldsymbol{g}_{j\ell}\right] = \sum_{k\in[d]}\sum_{\ell\in[d]} \boldsymbol{w}_k \boldsymbol{w}_\ell \, \mathbf{E}[\boldsymbol{g}_{ik}\boldsymbol{g}_{j\ell}] = 0 \qquad \forall i \neq j,$$

where the last equality follows from independence between $\boldsymbol{g}_i$ and $\boldsymbol{g}_j$. ■

A.2 Proof of Theorem 1 and its implications

In this subsection, we will show that there exists a choice of attention architecture Ψ which allows the model to distinguish inter-class edges and intra-class edges. In particular, we construct an explicit instance Ψ and show that it ignores all "unimportant" edges and keeps only "important" edges. We restate the theorem for convenience.

Theorem. *Suppose that $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$. Then, there exists a choice of attention architecture Ψ such that with a probability of at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim \mathit{CSBM}(n, p, q, \boldsymbol{\mu}, \sigma^2)$ it holds that Ψ separates intra-class edges from inter-class edges.*

Proof: We consider as an ansatz the following two layer architecture Ψ.

$$\tilde{\boldsymbol{w}} \overset{\text{def}}{=} \operatorname{sign}(p-q)\frac{\boldsymbol{\mu}}{\|\boldsymbol{\mu}\|_2}, \qquad \mathbf{S} \overset{\text{def}}{=} \begin{bmatrix} 1 & 1 \\ -1 & -1 \\ 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad \boldsymbol{r} \overset{\text{def}}{=} R \cdot [1 \quad 1 \quad -1 \quad -1]$$

where $R > 0$ is a scaling parameter to be determined later. The output of the attention model is defined as

$$\Psi(\mathbf{X}_i, \mathbf{X}_j) := \boldsymbol{r} \cdot \operatorname{LeakyRelu}\left(\mathbf{S} \cdot \begin{bmatrix} \tilde{\boldsymbol{w}}^T \mathbf{X}_i \\ \tilde{\boldsymbol{w}}^T \mathbf{X}_j \end{bmatrix}\right).$$

We will assume that $p \geq q$ and treat $\operatorname{sign}(0) = 1$. The result for $p < q$ follows similarly. Denote the input of $\operatorname{LeakyRelu}(\cdot)$ by $\Delta_{ij} \overset{\text{def}}{=} \mathbf{S} \cdot \begin{bmatrix} \tilde{\boldsymbol{w}}^T \mathbf{X}_i \\ \tilde{\boldsymbol{w}}^T \mathbf{X}_j \end{bmatrix} \in \mathbb{R}^4$, and note that for $t \in [4]$, we have $(\Delta_{ij})_t = \mathbf{S}_{t,1}\tilde{\boldsymbol{w}}^T \mathbf{X}_i + \mathbf{S}_{t,2}\tilde{\boldsymbol{w}}^T \mathbf{X}_j$. Recall that the random variable $(\Delta_{ij})_t = \mathbf{S}_{t,1}\tilde{\boldsymbol{w}}^T \mathbf{X}_i + \mathbf{S}_{t,2}\tilde{\boldsymbol{w}}^T \mathbf{X}_j$ is distributed as follows:

$$(\Delta_{ij})_t = \mathbf{S}_{t,1}\tilde{\boldsymbol{w}}^T \mathbf{X}_i + \mathbf{S}_{t,2}\tilde{\boldsymbol{w}}^T \mathbf{X}_j \sim \begin{cases} N((\mathbf{S}_{t,1} + \mathbf{S}_{t,2})\tilde{\boldsymbol{w}}^T \boldsymbol{\mu},\ \|\mathbf{S}_t\|^2\sigma^2) & \text{if } i,j \in C_1 \\ N((\mathbf{S}_{t,1} - \mathbf{S}_{t,2})\tilde{\boldsymbol{w}}^T \boldsymbol{\mu},\ \|\mathbf{S}_t\|^2\sigma^2) & \text{if } i \in C_1,\ j \in C_0 \\ N(-(\mathbf{S}_{t,1} - \mathbf{S}_{t,2})\tilde{\boldsymbol{w}}^T \boldsymbol{\mu},\ \|\mathbf{S}_t\|^2\sigma^2) & \text{if } i \in C_0,\ j \in C_1 \\ N(-(\mathbf{S}_{t,1} + \mathbf{S}_{t,2})\tilde{\boldsymbol{w}}^T \boldsymbol{\mu},\ \|\mathbf{S}_t\|^2\sigma^2) & \text{if } i,j \in C_0 \end{cases}.$$

We work on each of the four coordinates separately. Assume $t = 1$. In such a case, we have that

$$(\Delta_{ij})_1 \sim \begin{cases} N(2\|\boldsymbol{\mu}\|_2,\ 2\sigma^2) & \text{if } i,j \in C_1 \\ N(0,\ 2\sigma^2) & \text{if } i \in C_1,\ j \in C_0 \\ N(0,\ 2\sigma^2) & \text{if } i \in C_0,\ j \in C_1 \\ N(-2\|\boldsymbol{\mu}\|_2,\ 2\sigma^2) & \text{if } i,j \in C_0 \end{cases}.$$

Using our results for the LeakyRelu concentration in Lemma A.6 and our assumption on the norm of $\boldsymbol{\mu}$, we have that with a probability of at least $1 - o_n(1)$,

$$\operatorname{LeakyRelu}((\Delta_{ij})_1) = \begin{cases} 2\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i,j \in C_1 \\ \pm 2C\sigma\sqrt{\log n} & \text{if } i \in C_1,\ j \in C_0 \\ \pm 2C\sigma\sqrt{\log n} & \text{if } i \in C_0,\ j \in C_1 \\ -2\beta\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i,j \in C_0 \end{cases}.$$

Using a similar argument we get

$$\mathrm{LeakyRelu}((\Delta_{ij})_2) = \begin{cases} -2\beta\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i, j \in C_1 \\ \pm 2C\sigma\sqrt{\log n} & \text{if } i \in C_1,\ j \in C_0 \\ \pm 2C\sigma\sqrt{\log n} & \text{if } i \in C_0,\ j \in C_1 \\ 2\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i, j \in C_0 \end{cases},$$

$$\mathrm{LeakyRelu}((\Delta_{ij})_3) = \begin{cases} \pm 2C\sigma\sqrt{\log n} & \text{if } i, j \in C_1 \\ 2\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i \in C_1,\ j \in C_0 \\ -2\beta\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i \in C_0,\ j \in C_1 \\ \pm 2C\sigma\sqrt{\log n} & \text{if } i, j \in C_0 \end{cases},$$

$$\mathrm{LeakyRelu}((\Delta_{ij})_4) = \begin{cases} \pm 2C\sigma\sqrt{\log n} & \text{if } i, j \in C_1 \\ -2\beta\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i \in C_1,\ j \in C_0 \\ 2\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \text{if } i \in C_0,\ j \in C_1 \\ \pm 2C\sigma\sqrt{\log n} & \text{if } i, j \in C_0 \end{cases}.$$

Applying a union bound over the four coordinates of the vector Δ_{ij}, we get that the above event holds with probability at least $1 - o_n(1)$ for all t.

Next, we examine the second layer of the architecture. Suppose $i, j \in C_1$ so that

$$\mathrm{LeakyRelu}(\Delta_{ij}) = \begin{bmatrix} 2\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & -2\beta\|\boldsymbol{\mu}\|_2(1 \pm o(1)) & \pm 2C\sigma\sqrt{\log n} & \pm 2C\sigma\sqrt{\log n} \end{bmatrix}.$$

Then,

$$\boldsymbol{r} \cdot \mathrm{LeakyRelu}(\Delta_{ij}) = 2R\|\boldsymbol{\mu}\|_2(1-\beta)(1 \pm o(1)) \pm 4RC\sigma\sqrt{\log n} = 2R\|\boldsymbol{\mu}\|_2(1-\beta)(1 \pm o(1)).$$

By applying a similar reasoning to the over pairs

$$\boldsymbol{r} \cdot \mathrm{LeakyRelu}(\Delta_{ij}) = \begin{cases} 2R\|\boldsymbol{\mu}\|_2(1-\beta)(1 \pm o(1)) & \text{if } i, j \in C_1 \\ 2R\|\boldsymbol{\mu}\|_2(1-\beta)(1 \pm o(1)) & \text{if } i, j \in C_0 \\ -2R\|\boldsymbol{\mu}\|_2(1-\beta)(1 \pm o(1)) & \text{if } i \in C_1,\ j \in C_0 \\ -2R\|\boldsymbol{\mu}\|_2(1-\beta)(1 \pm o(1)) & \text{if } i \in C_0,\ j \in C_1 \end{cases},$$

and the proof is complete. ■

Next, we define a high probability event under which we can obtain some interesting corollaries.

Definition A.10. *Event $\boldsymbol{\mathcal{E}}^*$ is the intersection of the following events over the randomness of $\mathbf{A}$ and $\{\epsilon_i\}_i$ and $\mathbf{X}_i$,*

1. *$\boldsymbol{\mathcal{E}}_1$ is the event that $|C_0| = \frac{n}{2} \pm O(\sqrt{n \log n})$ and $|C_1| = \frac{n}{2} \pm O(\sqrt{n \log n})$.*
2. *$\boldsymbol{\mathcal{E}}_2$ is the event that for each $i \in [n]$, $\mathbf{D}_{ii} = \frac{n(p+q)}{2}\left(1 \pm \frac{10}{\sqrt{\log n}}\right)$.*
3. *$\boldsymbol{\mathcal{E}}_3$ is the event that for each $i \in [n]$, $|C_0 \cap N_i| = \mathbf{D}_{ii} \cdot \frac{(1-\epsilon_i)p+\epsilon_i q}{p+q}\left(1 \pm \frac{10}{\sqrt{\log n}}\right)$ and $|C_1 \cap N_i| = \mathbf{D}_{ii} \cdot \frac{(1-\epsilon_i)q+\epsilon_i p}{p+q}\left(1 \pm \frac{10}{\sqrt{\log n}}\right)$.*
4. *$\boldsymbol{\mathcal{E}}_4$ is the event that for each $i \in [n]$, $\left|\tilde{\boldsymbol{w}}^T\mathbf{X}_i - \mathbf{E}\left[\tilde{\boldsymbol{w}}^T\mathbf{X}_i\right]\right| \leq 10\sigma\sqrt{\log n}$.*

The next lemma is a straightforward application of Chernoff bound and a union bound (originally proved in Baranwal et al. (2021))

Lemma A.11. *With probability at least $1 - 1/\mathrm{poly}(n)$ event $\boldsymbol{\mathcal{E}}^*$ holds.*

We present two corollaries of Theorem A.2. The first is regarding the values of γ_{ij}.

Corollary 8. *Suppose that $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$. Then there exists a choice of attention architecture Ψ such that with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim$ CSBM$(n, p, q, \boldsymbol{\mu}, \sigma^2)$ it holds that*

1. *If $p \geq q$, then $\gamma_{ij} = \frac{2}{np}(1 \pm o_n(1))$ if (i,j) is an intra-class edge and $\gamma_{ij} = o(\frac{1}{n(p+q)})$ otherwise;*
2. *If $p < q$, then $\gamma_{ij} = \frac{2}{nq}(1 \pm o_n(1))$ if (i,j) is an inter-class edge and $\gamma_{ij} = o(\frac{1}{n(p+q)})$ otherwise.*

Proof: The proof is straightforward by considering the cases $p \geq q$ and $p < q$ separately. Using the attention architecture given in Theorem A.2, the definition of the attention coefficients, the high probability event in Lemma A.11, and picking R such that $1/R = \omega(\sigma\sqrt{\log n})$ and $1/R = o(\|\boldsymbol{\mu}\|_2)$, we obtain the claimed results. ∎

Next, we show that the model separates the nodes for any choice of p, q satisfying Assumption 1.

Corollary 9. *Suppose that $\|\boldsymbol{\mu}\|_2 = \omega(\sigma\sqrt{\log n})$. Then, there exists a choice of attention architecture Ψ such that with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim \mathsf{CSBM}(n, p, q, \boldsymbol{\mu}, \sigma^2)$, GAT separates the data for any p, q satisfying Assumption 1.*

Proof: We prove the case $p \geq q$ and the case $p < q$ follows analogously. Consider the same ansatz described in Theorem A.2 (set R such that $1/R = \omega(\sigma\sqrt{\log n})$ and $1/R = o(\|\boldsymbol{\mu}\|_2)$). Assume that $i \in C_1$ (the case for $i \in C_0$ follows similarly), and let

$$\hat{\boldsymbol{x}}_i \overset{\text{def}}{=} \sum_{j \in N_i} \gamma_{ij} \tilde{\boldsymbol{w}}^T \mathbf{X}_j.$$

We would like to compute the conditional mean and variance of $\hat{\boldsymbol{x}}_i$ given $\boldsymbol{\mathcal{E}}^*$. By using Corollary 8 we have

$$\begin{aligned}
\mathbf{E}\left[\sum_{j \in N_i} \gamma_{ij} \tilde{\boldsymbol{w}}^T \mathbf{X}_j \middle| \boldsymbol{\mathcal{E}}^*\right] &= \mathbf{E}\left[\sum_{j \in C_0 \cap N_i} \gamma_{ij} \tilde{\boldsymbol{w}}^T \mathbf{X}_j + \sum_{j \in C_1 \cap N_i} \gamma_{ij} \tilde{\boldsymbol{w}}^T \mathbf{X}_j \middle| \boldsymbol{\mathcal{E}}^*\right] \\
\leq\ & |C_1 \cap N_i| \left(\frac{2}{np}(1 \pm o(1)) \left(\|\boldsymbol{\mu}\|_2 + 10\sigma\sqrt{\log n}\right)\right) + |C_0 \cap N_i| \left(o\left(\frac{1}{n(p+q)}\right) \left(-\|\boldsymbol{\mu}\|_2 + 10\sigma\sqrt{\log n}\right)\right) \\
=\ & \|\boldsymbol{\mu}\|_2(1 \pm o(1)) + 10\sigma\sqrt{\log n} - \frac{nq(1 \pm o(1))}{2 \cdot \omega(n(p+q))} \left(\|\boldsymbol{\mu}\|_2 - 10\sigma\sqrt{\log n}\right) \\
=\ & \|\boldsymbol{\mu}\|_2(1 \pm o(1)).
\end{aligned}$$

Similarly,

$$\begin{aligned}
\mathbf{E}\left[\sum_{j \in N_i} \gamma_{ij} \tilde{\boldsymbol{w}}^T \mathbf{X}_j \middle| \boldsymbol{\mathcal{E}}^*\right] &\geq \|\boldsymbol{\mu}\|_2(1 \pm o(1)) - 10\sigma\sqrt{\log n} - \frac{nq(1 \pm o(1))}{2 \cdot \omega(n(p+q))} \left(\|\boldsymbol{\mu}\|_2 + 10\sigma\sqrt{\log n}\right) \\
&= \|\boldsymbol{\mu}\|_2(1 \pm o(1)).
\end{aligned}$$

Using the same reasoning, we get for $i \in C_0$, $\mathbf{E}[\hat{\boldsymbol{x}}_i | \boldsymbol{\mathcal{E}}^*] = -\|\boldsymbol{\mu}\|_2(1 \pm o(1))$.

Next, we claim that for each $i \in [n]$ the random variable $\hat{\boldsymbol{x}}_i$ given $\boldsymbol{\mathcal{E}}^*$ is sub-Gaussian with a small sub-Gaussian constant compared to the above expectation.

Lemma A.12. *Conditioned on $\boldsymbol{\mathcal{E}}^*$, the random variables $\hat{\boldsymbol{x}}_i$ for $i \in [n]$ are sub-Gaussian with sub-Gaussian parameter $\tilde{\sigma}^2 = O\left(\frac{\sigma^2}{np}\right)$.*

Proof: Fix an arbitrary $i \in [n]$. In order to obtain a sub-Gaussian parameter of $\hat{\boldsymbol{x}}_i$ conditioned on the event $\boldsymbol{\mathcal{E}}^*$, we will use concentration of Lipschitz functions of Gaussian random variables, see, e.g., Theorem 5.2.2 in Vershynin (2018). In particular, we will show that there is a Lipschitz function $f_i : \mathbb{R}^n \to \mathbb{R}$ such that the distribution of $f_i(\boldsymbol{v})$ for $\boldsymbol{v} \sim N(0, \mathbf{I}_n)$ is the same as the conditional distribution of $\hat{\boldsymbol{x}}_i$ conditioned on the event $\boldsymbol{\mathcal{E}}^*$. In what follows we construct the function f_i in a series of steps.

Let us write $\mathbf{X}_i = (2\epsilon_i - 1)\boldsymbol{\mu} + \sigma \boldsymbol{g}_i$ where $\boldsymbol{g}_i \sim N(0, \mathbf{I}_d)$, $\epsilon_i = 0$ if $i \in C_0$ and $\epsilon_i = 1$ if $i \in C_1$. Because $\tilde{\boldsymbol{w}} = \boldsymbol{\mu}/\|\boldsymbol{\mu}\|_2$ we have $\tilde{\boldsymbol{w}}^T \mathbf{X}_i = (2\epsilon_i - 1)\|\boldsymbol{\mu}\|_2 + \sigma \tilde{\boldsymbol{w}}^T \boldsymbol{g}_i$. We will consider a random vector $\boldsymbol{v} \in \mathbb{R}^n$ whose jth coordinate $\boldsymbol{v}_j$ has the same distribution as $\tilde{\boldsymbol{w}}^T \boldsymbol{g}_j$. By Observation A.9, $\boldsymbol{v} \sim N(0, \mathbf{I}_n)$.

Note that the event $\boldsymbol{\mathcal{E}}^*$ (more specifically, the event $\boldsymbol{\mathcal{E}}_4$) induces a transformation which transforms the isotropic Gaussian random vector $[\tilde{\boldsymbol{w}}^T \boldsymbol{g}_j]_{j \in [n]}$ to a vector of truncated Gaussian random variables. This is because the event

$\boldsymbol{\mathcal{E}}^*$ requires that $\left|\tilde{\boldsymbol{w}}^T\mathbf{X}_j - \mathbf{E}\left[\tilde{\boldsymbol{w}}^T\mathbf{X}_j\right]\right| \leq 10\sigma\sqrt{\log n}$ for all $j \in [n]$, but since $\tilde{\boldsymbol{w}}^T\mathbf{X}_j - \mathbf{E}\left[\tilde{\boldsymbol{w}}^T\mathbf{X}_j\right] = \sigma\tilde{\boldsymbol{w}}^T\boldsymbol{g}_j$, this is equivalent to requiring that $|\tilde{\boldsymbol{w}}^T\boldsymbol{g}_j| \leq 10\sqrt{\log n}$ for all $j \in [n]$. Therefore, conditioned on the event $\boldsymbol{\mathcal{E}}^*$, for each $j \in [n]$ the random variable $\tilde{\boldsymbol{w}}^T\boldsymbol{g}_j$ follows a truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$. Let $\bar{\boldsymbol{v}} \in \mathbb{R}^n$ denote the random vector whose jth coordinate $\bar{\boldsymbol{v}}_j$ has a truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$. We show that $\bar{\boldsymbol{v}}$ may be obtained from $\boldsymbol{v}$ via a push forward mapping M:

$$\bar{\boldsymbol{v}} = M(\boldsymbol{v}) \overset{\text{def}}{=} [\tau(\boldsymbol{v}_1), \tau(\boldsymbol{v}_2), \ldots, \tau(\boldsymbol{v}_n)]^T \tag{16}$$

where $\tau(x) \overset{\text{def}}{=} \Phi^{-1}((1-2c)\Phi(x)+c)$ for $c = \Phi(-10\sqrt{\log n})$. The following claim shows that $\tau(\boldsymbol{v}_j)$ indeed follows the truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$.

Claim A.13. *Assume that $v \sim N(0,1)$. Then, $\tau(v)$ follows the truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$.*

Proof: Let $\bar{v}$ be a random variable that follows the truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$. Its cumulative distribution function is given by $\Psi(x) = (\Phi(x) - c)/(1-2c)$ where $c = \Phi(-10\sqrt{\log n})$. The function $\Psi : [-10\sqrt{\log n}, 10\sqrt{\log n}] \to [0,1]$ is bijective and has inverse $\Psi^{-1} : [0,1] \to [-10\sqrt{\log n}, 10\sqrt{\log n}]$. In particular, if $\Psi(x) = u$ for some $x \in [-10\sqrt{\log n}, 10\sqrt{\log n}]$ and $u \in [0,1]$, then we know that $x = \Psi^{-1}(u) = \Phi^{-1}((1-2c)u+c)$. By the inverse transform method, if u follows a uniform distribution over the interval $[0,1]$, then $\Psi^{-1}(u)$ follows the truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$. Let $v \sim N(0,1)$, then $\Phi(v)$ is uniform over $[0,1]$, and hence $\tau(v) = \Phi^{-1}((1-2c)\Phi(v)+c) = \Psi^{-1}(\Phi(v))$ follows the truncated Gaussian distribution over the interval $[-10\sqrt{\log n}, 10\sqrt{\log n}]$. ■

It turns out that the push forward mapping $M : \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz with constant 1.

Claim A.14. *The mapping M given by equation 16 has Lipschitz constant 1.*

Proof: We show that the coordinate transform τ is Lipschitz which implies the result. Because the cumulative distribution function Φ is differentiable and bijective, the derivative of the inverse Φ^{-1} is given by the inverse function rule: $\frac{d}{dx}\Phi^{-1}(x) = 1/(\phi(\Phi^{-1}(x)))$, where $\phi(x)$ denote the standard Gaussian PDF. Apply the chain rule and the inverse function rule we get that

$$\frac{d}{dx}\tau(x) = \frac{d}{dx}\Big[\Phi^{-1}((1-2c)\Phi(x)+c)\Big] = \frac{(1-2c)\phi(x)}{\phi\Big(\Phi^{-1}((1-2c)\Phi(x)+c)\Big)} \leq \frac{(1-2c)\phi(x)}{\phi(x)} < 1. \tag{17}$$

In order to see the second last inequality, let us consider the following two cases.

Case 1: $x \geq 0$. In this case, we have that $\frac{1}{2} \leq \Phi(x) \leq 1$ and

$$\frac{1}{2} \leq \Phi(x) \leq 1 \iff 1 - 2\Phi(x) \leq 0 \iff c(1-2\Phi(x)) \leq 0 \iff (1-2c)\Phi(x) + c \leq \Phi(x).$$

Moreover, one easily verifies that

$$(1-2c)\Phi(x) + c \geq \frac{1}{2} \iff \Phi(x) \geq \frac{1}{2}.$$

Therefore, since $x \geq 0$, we have that $\frac{1}{2} \leq (1-2c)\Phi(x) + c \leq \Phi(x)$, which implies

$$0 \leq \Phi^{-1}((1-2c)\Phi(x)+c) \leq \Phi^{-1}(\Phi(x)) = x,$$

and hence $\phi(\Phi^{-1}((1-2c)\Phi(x)+c)) \geq \phi(x)$, proving the second last inequality of equation 17.

Case 2: $x \leq 0$. In this case, we have that $0 \leq \Phi(x) \leq \frac{1}{2}$. The result is shown by following the same steps as above.

It follows from equation 17 that the function τ has Lipschitz constant 1. The Lipschitz constant of M is obtained by noticing that

$$\|M(\boldsymbol{u}) - M(\boldsymbol{v})\|_2^2 = \sum_{j=1}^{n}(\tau(\boldsymbol{u}_j) - \tau(\boldsymbol{v}_j))^2 \leq \sum_{j=1}^{n}(\boldsymbol{u}_j - \boldsymbol{v}_j)^2 = \|\boldsymbol{u} - \boldsymbol{v}\|_2^2.$$

■

So far we have showed that $M(\boldsymbol{v})$ has the same distribution as $[\tilde{\boldsymbol{w}}^T \boldsymbol{g}_j]_{j\in[n]}$ conditioned on the event $\boldsymbol{\mathcal{E}}^*$. Moreover, M has Lipschitz constant $L_M = 1$. Now, consider the function $l : \mathbb{R}^n \to \mathbb{R}^n$ defined by

$$l(\bar{\boldsymbol{v}}) \stackrel{\text{def}}{=} \Big[(2\epsilon_j - 1)\|\boldsymbol{\mu}\|_2 + \sigma\bar{\boldsymbol{v}}_j\Big]_{j\in[n]}.$$

It is straightforward to see that the Lipschitz constant of l is $L_l = \sigma$, since

$$\|l(\bar{\boldsymbol{v}}) - l(\bar{\boldsymbol{v}}')\|_2 = \left\| \begin{bmatrix} \vdots \\ (2\epsilon_j - 1)\|\boldsymbol{\mu}\|_2 + \sigma\bar{\boldsymbol{v}}_j \\ \vdots \end{bmatrix}_{j\in[n]} - \begin{bmatrix} \vdots \\ (2\epsilon_j - 1)\|\boldsymbol{\mu}\|_2 + \sigma\bar{\boldsymbol{v}}'_j \\ \vdots \end{bmatrix}_{j\in[n]} \right\|_2 = \sigma\|\bar{\boldsymbol{v}} - \bar{\boldsymbol{v}}'\|_2.$$

In addition, since $\tilde{\boldsymbol{w}}^T\mathbf{X}_j = (2\epsilon_i - 1)\|\boldsymbol{\mu}\|_2 + \sigma\tilde{\boldsymbol{w}}^T\boldsymbol{g}_j$ for $j \in [n]$, we see that $l(M(\boldsymbol{v}))$ has the same distribution as $[\tilde{\boldsymbol{w}}^T\mathbf{X}_j]_{j\in[n]}$ conditioned on the event $\boldsymbol{\mathcal{E}}^*$. For $j \in [n]$ let $\tilde{\boldsymbol{w}}^T\mathbf{X}_j|\boldsymbol{\mathcal{E}}^*$ denote the random variable which follows the conditional distribution of $\tilde{\boldsymbol{w}}^T\mathbf{X}_j$ conditioned on the event $\boldsymbol{\mathcal{E}}^*$, and similarly let $\hat{\boldsymbol{x}}_i|\boldsymbol{\mathcal{E}}^*$ denote the random variable which follows the conditional distribution of $\hat{\boldsymbol{x}}_i$ conditioned on the event $\boldsymbol{\mathcal{E}}^*$. Because the unconditioned random variable $\hat{\boldsymbol{x}}_i$ is obtained as a function of $[\tilde{\boldsymbol{w}}^T\mathbf{X}_j]_{j\in[n]}$:

$$\hat{\boldsymbol{x}}_i = \sum_{i\in N_i} \gamma_{ij}\Big([\tilde{\boldsymbol{w}}^T\mathbf{X}_j]_{j\in[n]}\Big) \cdot \tilde{\boldsymbol{w}}^T\mathbf{X}_j,$$

it follows that

$$f_i(\boldsymbol{v}) \stackrel{\text{def}}{=} \sum_{j\in N_i} \gamma_{ij}(l(M(\boldsymbol{v}))) \cdot [l(M(\boldsymbol{v}))]_j \;=\; \sum_{j\in N_i} \gamma_{ij}\Big([\tilde{\boldsymbol{w}}^T\mathbf{X}_j|\boldsymbol{\mathcal{E}}^*]_{j\in[n]}\Big) \cdot \tilde{\boldsymbol{w}}^T\mathbf{X}_j|\boldsymbol{\mathcal{E}}^* \;=\; \hat{\boldsymbol{x}}_i|\boldsymbol{\mathcal{E}}^*,$$

where the second and the third equalities denote equality in distribution. As a technical remark, in order for $f_i(\boldsymbol{v})$ and $\hat{\boldsymbol{x}}_i|\boldsymbol{\mathcal{E}}^*$ to have identical distributions, we need to consider both distributions conditioning on the events $\boldsymbol{\mathcal{E}}_1, \boldsymbol{\mathcal{E}}_2, \boldsymbol{\mathcal{E}}_3$. These events are only concerned with the graph structure and do not affect the Gaussian distributions of $[\tilde{\boldsymbol{w}}^T\boldsymbol{g}_j]_{j\in[n]}$ or $\boldsymbol{v}$. For notational simplicity we omit conditioning on these events explicitly. We proceed the proof with the understanding that we are to establish distributional equivalence between $\hat{\boldsymbol{x}}_i|\boldsymbol{\mathcal{E}}_4$ and $f_i(\boldsymbol{v})$ under the event that $\boldsymbol{\mathcal{E}}_1, \boldsymbol{\mathcal{E}}_2, \boldsymbol{\mathcal{E}}_3$ already hold. This is without loss of generality and we will explain the reason when the conditions are used later in the proof.

It left to obtain a Lipschitz constant of f_i. We see that the function f_i is the composition $f_i = h_i \circ l \circ M$ where

$$h_i(\boldsymbol{x}) \stackrel{\text{def}}{=} \sum_{j\in N_i} \gamma_{ij}(\boldsymbol{x}) \cdot \boldsymbol{x}_j.$$

Therefore, the Lipschitz constant of f_i is obtained by $L_{f_i} = L_{h_i}L_lL_M = \sigma L_{h_i}$ where L_{h_i} is the Lipschitz constant of h_i. In what follows we compute L_{h_i}. The domain of the function h_i is the range $\mathcal{R}$ of the composition $l \circ M$. We will show that h_i is Lipschitz over $\mathcal{R}$. Let us assume without loss of generality that $i \in C_0$ (the case for $i \in C_1$ yields the same result and is obtained identically). By the definition of M and l, we know that the event $\boldsymbol{\mathcal{E}}_4$ identifies a bounded subspace in $\mathbb{R}^n$ which is essentially the set $\mathcal{R}$. Moreover, since the events $\boldsymbol{\mathcal{E}}_1, \boldsymbol{\mathcal{E}}_2, \boldsymbol{\mathcal{E}}_3$ do not affect the distribution of the Gaussian random variables and hence we may assume without loss of generality that these events hold (otherwise, one can obtain identical result by carrying out the same series of computations and then apply the conditions of events $\boldsymbol{\mathcal{E}}_1, \boldsymbol{\mathcal{E}}_2, \boldsymbol{\mathcal{E}}_3$). We know from Corollary 8 that under the event $\boldsymbol{\mathcal{E}}^*$ we have $\gamma_{ij}(\boldsymbol{x}) = \frac{2}{np}(1 \pm o(1))$ if $j \in C_0$ and

$\gamma_{ij}(\boldsymbol{x}) = \frac{2}{np}\exp(-\Theta(R\|\boldsymbol{\mu}\|_2))(1\pm o(1))$ if $j \in C_1$. Recall that R satisfies $R\|\boldsymbol{\mu}\|_2 = \omega(1)$, we get

$$
\begin{aligned}
|h_i(\boldsymbol{x}) - h_i(\boldsymbol{x}')| &= \left| \sum_{j\in N_i\cap C_0} \frac{2(1\pm o(1))}{np}(\boldsymbol{x}_j - \boldsymbol{x}_j') + \sum_{j\in N_i\cap C_1} \frac{2(1\pm o(1))}{np}\cdot e^{-\Theta(\|\boldsymbol{\mu}\|_2)}(\boldsymbol{x}_j - \boldsymbol{x}_j') \right| \\
&= \left| \begin{bmatrix} \frac{2}{np}(1\pm o(1)) & \text{if } j\in N_i\cap C_0 \\ \frac{2}{np}\exp(-\Theta(R\|\boldsymbol{\mu}\|_2))(1\pm o(1)) & \text{if } j\in N_i\cap C_1 \\ 0 & \text{if } j\notin N_i \end{bmatrix}^T_{j\in[n]} (\boldsymbol{x}-\boldsymbol{x}') \right| \\
&\leq \left\| \begin{bmatrix} \frac{2}{np}(1\pm o(1)) & \text{if } j\in N_i\cap C_0 \\ \frac{2}{np}\exp(-\Theta(R\|\boldsymbol{\mu}\|_2))(1\pm o(1)) & \text{if } j\in N_i\cap C_1 \\ 0 & \text{if } j\notin N_i \end{bmatrix}_{j\in[n]} \right\|_2 \|\boldsymbol{x}-\boldsymbol{x}'\|_2 \\
&\leq \sqrt{\frac{2}{np}}(1+o(1))\,\|\boldsymbol{x}-\boldsymbol{x}'\|_2
\end{aligned}
$$

This shows that the Lipschitz constant of h_i over $\mathcal{R}$ satisfies $L_{h_i} = O\left(\frac{1}{\sqrt{np}}\right)$. Therefore, the Lipschitz constant of f_i is $L_{f_i} = \sigma L_{h_i} = O\left(\frac{\sigma}{\sqrt{np}}\right)$. This allows us to apply the Gaussian concentration result (see Theorem 5.2.2 in Vershynin (2018)) to the random variable $f_i(\boldsymbol{v})$ and get that the sub-Gaussian parameter of $f_i(\boldsymbol{v})$ is $\tilde{\sigma}^2 = L_{f_i}^2 = O\left(\frac{\sigma^2}{np}\right)$. Since the random variable $\hat{\boldsymbol{x}}_i$ conditioned on $\boldsymbol{\mathcal{E}}^*$ has the same distribution of $f_i(\boldsymbol{v})$, its sub-Gaussian parameter is also $O\left(\frac{\sigma^2}{np}\right)$. The result holds for all $i\in[n]$ because our choice of i was arbitrary. ■

Now, we have all the tools to finish the proof of the theorem. We bound the probability of misclassification

$$\mathbf{Pr}\left[\max_{i\in C_0}\hat{\boldsymbol{x}}_i \geq 0\right] \leq \mathbf{Pr}\left[\max_{i\in C_0}\hat{\boldsymbol{x}}_i > t + \mathbf{E}[\hat{\boldsymbol{x}}_i]\right],$$

for $t < |\mathbf{E}[\hat{\boldsymbol{x}}_i]| = \|\boldsymbol{\mu}\|_2(1\pm o(1))$. By Lemma A.12, picking $t = \Theta\left(\sigma\sqrt{\log|C_0|}\right)$ and applying Lemma A.2 implies that the above probability is $1/\mathrm{poly}(n)$.

Similarly for class C_1 we have that the misclassification probability is

$$\mathbf{Pr}\left[\min_{i\in C_1}\hat{\boldsymbol{x}}_i \leq 0\right] = \mathbf{Pr}\left[-\max_{i\in C_1}(-\hat{\boldsymbol{x}}_i) \leq 0\right] = \mathbf{Pr}\left[\max_{i\in C_1}(-\hat{\boldsymbol{x}}_i) \geq 0\right] \leq \mathbf{Pr}\left[\max_{i\in C_1}-\hat{\boldsymbol{x}}_i > t - \mathbf{E}[\hat{\boldsymbol{x}}_i]\right],$$

for $t < \mathbf{E}[\hat{\boldsymbol{x}}_i]$. Picking $t = \Theta\left(\sigma\sqrt{\log|C_1|}\right)$ and applying Lemma A.2 and a union bound over the misclassification probabilities of both classes conclude the proof of the corollary. ■

A.3 PROOF OF PROPOSITION 4

We start by restating the proposition for convenience.

Proposition. *Suppose $\|\boldsymbol{\mu}\|_2 = \Omega(\sigma\sqrt{\log n})$, and let $(\mathbf{X},\cdot) \sim \mathit{CSBM}(n,\cdot,\cdot,\boldsymbol{\mu},\sigma^2)$. Then, the Bayes optimal classifier is realized by a linear classifier which achieves perfect node separability with probability at least $1-o_n(1)$ over $\mathbf{X}$.*

Proof: The proof follows by first applying Lemma A.7 to deduce that a linear classifier obtains the optimal performance and then applying Case (1) of Lemma A.8. ■

A.4 PROOF OF THEOREM 5

The goal of the attention mechanism is to separate the pairs of nodes $(\mathbf{X}_i, \mathbf{X}_j)$ based on whether i, j are in the same class or different classes. We say $i \sim j$ if both i and j are in the same class C_0 or C_1, and $i \nsim j$ otherwise. Let $\mathbf{X}'_{ij}$ denote the vector obtained as a result of concatenating $\mathbf{X}_i$ and $\mathbf{X}_j$, i.e., $\mathbf{X}'_{ij} = \begin{pmatrix}\mathbf{X}_i \\ \mathbf{X}_j\end{pmatrix}$. Then we would like to analyze all classifiers with the property

$$y = h(\mathbf{X}'_{ij}) = \begin{cases} 0 & i \nsim j \\ 1 & i \sim j \end{cases}.$$

To comment on the limitations of all such classifiers, it is sufficient to analyze the Bayes classifier for this data model, since by definition a Bayes classifier is optimal. The following lemma describes the optimal classifier for this classification task.

Lemma A.15. *The optimal (Bayes) classifier that serves as the attention mechanism for the pairs $\mathbf{X}'_{ij}$ is realized by the following function.*

$$h^*(\boldsymbol{x}) = \begin{cases} 0 & \textit{if } p\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) \leq q\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \\ 1 & \textit{otherwise} \end{cases}.$$

Proof: Recall that $|C_0| = |C_1| = \frac{n}{2}$. Hence, $\mathbf{X}'_{ij}$ is a uniform mixture of four $2d$-dimensional Gaussian distributions. Let $\boldsymbol{\mu}' = \begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu} \end{pmatrix}$ and $\boldsymbol{\nu}' = \begin{pmatrix} \boldsymbol{\mu} \\ -\boldsymbol{\mu} \end{pmatrix}$. Then the four Gaussian distributions are

$$\mathbf{X}'_{ij} \sim \begin{cases} N(\boldsymbol{\mu}', \sigma^2 I) & i \in C_0, j \in C_0 \\ N(-\boldsymbol{\mu}', \sigma^2 I) & i \in C_1, j \in C_1 \\ N(\boldsymbol{\nu}', \sigma^2 I) & i \in C_0, j \in C_1 \\ N(-\boldsymbol{\nu}', \sigma^2 I) & i \in C_1, j \in C_0 \end{cases}.$$

The optimal classifier is then given by

$$h^*(\boldsymbol{x}) = \operatorname*{arg\,max}_{c \in \{0,1\}} \mathbf{Pr}[y = c \mid \boldsymbol{x}].$$

Note that $\mathbf{Pr}[y = 0] = q$ and $\mathbf{Pr}[y = 1] = p$. Thus, by Bayes rule we obtain that

$$\mathbf{Pr}[y = c \mid \boldsymbol{x}] = \frac{\mathbf{Pr}[y = c] \cdot f_{\boldsymbol{x}|y}(\boldsymbol{x} \mid y = c)}{\mathbf{Pr}[y = 0] f_{\boldsymbol{x}|y=0}(\boldsymbol{x} \mid y = 0) + \mathbf{Pr}[y = 1] f_{\boldsymbol{x}|y=1}(\boldsymbol{x} \mid y = 1)} = \frac{1}{1 + \frac{\mathbf{Pr}[y=1-c] \cdot f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=1-c)}{\mathbf{Pr}[y=c] \cdot f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=c)}}.$$

Suppose that $\boldsymbol{x} = \mathbf{X}'_{ij}$ such that $i \nsim j$. Then $h^*(\boldsymbol{x}) = 0$ if and only if $\mathbf{Pr}[y = 0 \mid \boldsymbol{x}] \geq \frac{1}{2}$. Hence, for $c = 0$ we require $\frac{p \cdot f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=1)}{q \cdot f_{\boldsymbol{x}|y}(\boldsymbol{x}|y=0)} \leq 1$, which implies that $\frac{p}{q} \frac{\cosh\left(\frac{1}{\sigma^2}\boldsymbol{x}^T\boldsymbol{\mu}'\right)}{\cosh\left(\frac{1}{\sigma^2}\boldsymbol{x}^T\boldsymbol{\nu}'\right)} \leq 1$. Similarly, we obtain the reverse condition for $h^*(\boldsymbol{x}) = 1$. ■

The next theorem uses A.15 to precisely quantify the misclassification of node pairs that the attention mechanism exhibits. In particular, part 1 of the theorem implies that if $\|\boldsymbol{\mu}\|_2$ is linear in the standard deviation, σ, then with overwhelming probability the attention mechanism fails to distinguish a constant fraction of inter-edge pairs from the intra-edge pairs.

Furthermore, part 2 of the theorem characterizes a regime for the inter-edge probability q where the attention mechanism fails to distinguish at least one inter-edge node pair, by providing a lower bound on q in terms of the scale at which the distance between the means grows compared to the standard deviation σ. This aligns with the intuition that as we increase the distance between the means, it gets easier for the attention mechanism to correctly distinguish the node pairs. However, if q is also increased in the right proportion, or in other words, if the noise in the graph is increased, then the attention mechanism will still fail to correctly distinguish at least one of the inter-edge node pairs.

We restate the theorem for the readers' convenience.

Theorem (Restatement of Theorem 5)**.** *Assume that $q = \Omega(\frac{\log^2 n}{n})$ and let Ψ be any attention mechanism. Let $\Phi_{\mathrm{c}}(\cdot) \overset{\text{def}}{=} 1 - \Phi(\cdot)$, where Φ is the standard normal CDF. Then for any $K > 0$ if $\|\boldsymbol{\mu}\|_2 = K\sigma$ then we have:*

1. *For any $c > 0$, with probability at least $1 - O(n^{-c})$, the attention mechanism Ψ fails to distinguish at least $2\Phi_{\mathrm{c}}(K)^2$ fraction of the inter-edge pairs $(\mathbf{X}_i, \mathbf{X}_j), i \nsim j$ from the intra-edge pairs $(\mathbf{X}_i, \mathbf{X}_j), i \sim j$.*

2. *For any $\kappa > 1$ if $q > \frac{\kappa \log^2 n}{n\Phi_{\mathrm{c}}(K)^2}$, then with probability at least $1 - O\left(\frac{1}{n^{\frac{\kappa}{4}\Phi_{\mathrm{c}}(K)^2 \log n}}\right)$, at least one inter-edge pair is indistinguishable from the intra-edge pairs under Ψ.*

Proof: From A.15, we observe that for successful classification by the optimal classifier, we need

$$\begin{aligned} p\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) &\leq q\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \quad \text{for } i \nsim j, \\ p\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) &> q\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \quad \text{for } i \sim j. \end{aligned}$$

We will split the analysis into two cases. First, note that when $p \geq q$ we have for $i \nsim j$ that

$$p\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) \leq q\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \implies \cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) \leq \cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \implies |\boldsymbol{x}^T\boldsymbol{\mu}'| \leq |\boldsymbol{x}^T\boldsymbol{\nu}'|.$$

In the first implication, we used that $p \geq q$, while the second implication follows from the fact that $\cosh(a) \leq \cosh(b) \implies |a| \leq |b|$ for all $a, b \in \mathbb{R}$. Similarly, for $p < q$ we have for $i \sim j$ that

$$p\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) > q\cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \implies \cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\mu}'}{\sigma^2}\right) > \cosh\left(\frac{\boldsymbol{x}^T\boldsymbol{\nu}'}{\sigma^2}\right) \implies |\boldsymbol{x}^T\boldsymbol{\mu}'| > |\boldsymbol{x}^T\boldsymbol{\nu}'|.$$

Therefore, for each of the above cases, we can upper bound the probability for either $i \sim j$ or $i \nsim j$ that $\mathbf{X}'_{ij}$ is correctly classified, by the probability of the event $|\mathbf{X}'^T_{ij}\boldsymbol{\mu}'| \leq |\mathbf{X}'^T_{ij}\boldsymbol{\nu}'|$ or equivalently $|\mathbf{X}'^T_{ij}\boldsymbol{\mu}'| > |\mathbf{X}'^T_{ij}\boldsymbol{\nu}'|$. We focus on the former as the latter is equivalent and symmetric. Writing $\mathbf{X}_i = \boldsymbol{\mu} + \sigma\boldsymbol{g}_i$ and $\mathbf{X}_j = -\boldsymbol{\mu} + \sigma\boldsymbol{g}_j$, we have that for $i \in C_1$ and $j \in C_0$,

$$\begin{aligned} \mathbf{Pr}[h^*(\mathbf{X}'_{ij}) = 0] &\leq \mathbf{Pr}\left[|\mathbf{X}'^T_{ij}\boldsymbol{\mu}'| \leq |\mathbf{X}'^T_{ij}\boldsymbol{\nu}'|\right] \\ &= \mathbf{Pr}\left[|\mathbf{X}_i^T\boldsymbol{\mu} + \mathbf{X}_j^T\boldsymbol{\mu}| \leq |\mathbf{X}_i^T\boldsymbol{\mu} - \mathbf{X}_j^T\boldsymbol{\mu}|\right] \\ &= \mathbf{Pr}\left[\sigma|\boldsymbol{g}_i^T\boldsymbol{\mu} + \boldsymbol{g}_j^T\boldsymbol{\mu}| \leq |\pm 2\|\boldsymbol{\mu}\|_2^2 + \sigma\boldsymbol{g}_i^T\boldsymbol{\mu} - \sigma\boldsymbol{g}_j^T\boldsymbol{\mu}|\right] \\ &\leq \mathbf{Pr}\left[|\boldsymbol{g}_i^T\hat{\boldsymbol{\mu}} + \boldsymbol{g}_j^T\hat{\boldsymbol{\mu}}| - |\boldsymbol{g}_i^T\hat{\boldsymbol{\mu}} - \boldsymbol{g}_j^T\hat{\boldsymbol{\mu}}| \leq \frac{2\|\boldsymbol{\mu}\|_2}{\sigma}\right] \\ &= \mathbf{Pr}\left[|\boldsymbol{g}_i^T\hat{\boldsymbol{\mu}} + \boldsymbol{g}_j^T\hat{\boldsymbol{\mu}}| - |\boldsymbol{g}_i^T\hat{\boldsymbol{\mu}} - \boldsymbol{g}_j^T\hat{\boldsymbol{\mu}}| \leq 2K\right], \end{aligned}$$

where $\hat{\boldsymbol{\mu}} = \frac{\boldsymbol{\mu}}{\|\boldsymbol{\mu}\|_2}$. In the second to last step above, we used triangle inequality to pull $2\|\boldsymbol{\mu}\|_2^2$ outside the absolute value, while in the last equation we use $\|\boldsymbol{\mu}\|_2 = K\sigma$.

We now denote $z_i = \boldsymbol{g}_i^T\hat{\boldsymbol{\mu}}$ for all $i \in [n]$. Then the above probability is $\mathbf{Pr}[|z_i + z_j| - |z_i - z_j| \leq 2K]$, where $z_i, z_j \sim N(0, 1)$ are independent random variables. Note that we have

$$\begin{aligned} \mathbf{Pr}[h^*(\mathbf{X}'_{ij}) = 0] &\leq \mathbf{Pr}[|z_i + z_j| - |z_i - z_j| \leq 2K] \\ &= \mathbf{Pr}[|z_i + z_j| - |z_i - z_j| \leq 2K, |z_i| \leq K] + \mathbf{Pr}[|z_i + z_j| - |z_i - z_j| \leq 2K, |z_i| > K] \\ &= \mathbf{Pr}[|z_i| \leq K] + \Phi(K)\,\mathbf{Pr}[|z_i| > K]. \end{aligned} \tag{18}$$

To see how we obtain the last equation, observe that if $|z_i| \leq K$ then we have

$$\begin{aligned} |z_i + z_j| - |z_i - z_j| &= |z_i + z_j| - |z_j - z_i| \\ &\leq |z_i| + |z_j| - |z_j - z_i| && \text{by triangle inequality} \\ &\leq |z_i| + |z_j| - \big||z_j| - |z_i|\big| && \text{by reverse triangle inequality} \\ &\leq |z_i| + |z_j| - (|z_j| - |z_i|) = 2|z_i| \\ &\leq 2K, \end{aligned}$$

hence, $\mathbf{Pr}[|z_i + z_j| - |z_i - z_j| \leq 2K, |z_i| \leq K] = \mathbf{Pr}[|z_i| \leq K]$. On the other hand, for $|z_i| > K$, we look at each case, conditioned on the events $z_i > K$ and $z_i < -K$ for each of the four cases based on the signs of $z_i + z_j$ and $z_i - z_j$. We denote by E the event that $|z_i + z_j| - |z_i - z_j| \leq 2K$, and analyze the cases in detail.

Conditioned on $z_i < -K$:

$$\begin{aligned} &\mathbf{Pr}[E, z_i + z_j \geq 0, z_i - z_j \geq 0 \mid z_i < -K] = \mathbf{Pr}[z_j \leq z_i, z_j \geq -z_i \mid z_i < -K] = 0, \\ &\mathbf{Pr}[E, z_i + z_j \geq 0, z_i - z_j < 0 \mid z_i < -K] = \mathbf{Pr}[z_j > |z_i|, z_i \leq K \mid z_i < -K] = \Phi(z_i), \\ &\mathbf{Pr}[E, z_i + z_j < 0, z_i - z_j \geq 0 \mid z_i < -K] = \mathbf{Pr}[z_j < -|z_i|, z_i \geq -K \mid z_i < -K] = 0, \\ &\mathbf{Pr}[E, z_i + z_j < 0, z_i - z_j < 0 \mid z_i < -K] = \mathbf{Pr}[z_i < z_j < -z_i, z_j > -K \mid z_i < -K] = \Phi(K) - \Phi(z_i). \end{aligned}$$

The sum of the four probabilities in the above display is $\mathbf{Pr}[E \mid z_i < -K] = \Phi(K)$. Similarly, we analyze the other case.

Conditioned on $z_i > K$:

$$\begin{aligned}
&\mathbf{Pr}[E, z_i + z_j \geq 0, z_i - z_j \geq 0 \mid z_i > K] = \mathbf{Pr}[-z_i \leq z_j \leq z_i, z_j \leq K \mid z_i > K] = \Phi(K) - \Phi_{\mathrm{c}}(z_i),\\
&\mathbf{Pr}[E, z_i + z_j \geq 0, z_i - z_j < 0 \mid z_i > K] = \mathbf{Pr}[z_j > |z_i|, z_i \leq K \mid z_i > K] = 0,\\
&\mathbf{Pr}[E, z_i + z_j < 0, z_i - z_j \geq 0 \mid z_i > K] = \mathbf{Pr}[z_j < -|z_i|, z_i \geq -K \mid z_i > K] = \Phi_{\mathrm{c}}(z_i),\\
&\mathbf{Pr}[E, z_i + z_j < 0, z_i - z_j < 0 \mid z_i > K] = \mathbf{Pr}[z_j < -z_i, z_j > z_i \mid z_i > K] = 0.
\end{aligned}$$

The sum of the four probabilities above is $\mathbf{Pr}[E \mid z_i > K] = \Phi(K)$. Therefore, we obtain that

$$\mathbf{Pr}[|z_i + z_j| - |z_i - z_j| \leq 2K \mid |z_i| > K] = \Phi(K),$$

which justifies Equation 18.

Next, note that $\mathbf{Pr}[|z_i| \leq K] = \Phi(K) - \Phi_{\mathrm{c}}(K)$ and $\mathbf{Pr}[|z_i| > K] = 2\Phi_{\mathrm{c}}(K)$, so we have from Equation 18 that

$$\mathbf{Pr}[h^*(\mathbf{X}'_{ij}) = 0] \leq \Phi(K) - \Phi_{\mathrm{c}}(K) + 2\Phi_{\mathrm{c}}(K)\Phi(K) = 1 - 2\Phi_{\mathrm{c}}(K) + 2\Phi_{\mathrm{c}}(K)\Phi(K) = 1 - 2\Phi_{\mathrm{c}}(K)^2.$$

Thus, $\mathbf{X}'_{ij}$ is misclassified with probability at least $2\Phi_{\mathrm{c}}(K)^2$.

We will now construct sets of pairs with mutually independent elements, such that the union of those sets covers all inter-edge pairs. This will enable us to use a concentration argument that computes the fraction of the inter-edge pairs that are misclassified. Since the graph operations are permutation invariant, let us assume for simplicity that $C_0 = \{1, \ldots, \frac{n}{2}\}$ and $C_1 = \{\frac{n}{2} + 1, \ldots, n\}$ for an even number of nodes n. Also define the function

$$m(i, l) = \begin{cases} i + l & i + l \leq \frac{n}{2} \\ i + l - \frac{n}{2} & i + l > \frac{n}{2} \end{cases}.$$

We now construct the following sequence of sets for all $l \in \{0, \ldots, \frac{n}{2} - 1\}$:

$$S_l = \{(X_{m(i,l)}, X_{i+\frac{n}{2}}) \text{ for all } i \in C_0 \cap N_{m(i,l)}\}.$$

We now fix $l \in \{0, \ldots, \frac{n}{2} - 1\}$ and observe that the pairs in the set S_l are mutually independent. Define a Bernoulli random variable, β_i, to be the indicator that $(X_{m(i,l)}, X_{i+\frac{n}{2}})$ is misclassified. We have that $\mathbf{E}[\beta_i] \geq 2\Phi_{\mathrm{c}}(K)^2$. Note that the fraction of pairs in the set S_l that are misclassified is $\frac{1}{|S_l|}\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i$, which is a sum of independent Bernoulli random variables. Hence, by Hoeffding's inequality, we obtain

$$\mathbf{Pr}\left[\frac{1}{|S_l|}\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i \geq 2\Phi_{\mathrm{c}}(K)^2 - t\right] \geq 1 - \exp(-|S_l|t^2).$$

Since $p, q = \omega(\frac{\log^2 n}{n})$, we have by the Chernoff bound that with probability at least $1 - 1/\mathrm{poly}(n)$, $|S_l| = nq(1 \pm o_n(1))$ for all l. We now choose $t = \sqrt{\frac{C \log n}{|S_l|}} = o_n(1)$ to obtain that on the event where $|S_l| = nq(1 \pm o_n(1))$, we have the following for any large $C > 1$:

$$\mathbf{Pr}\left[\frac{1}{|S_l|}\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i \geq 2\Phi_{\mathrm{c}}(K)^2 - o_n(1)\right] \geq 1 - n^{-C}.$$

Following a union bound over all $l \in \{0, \ldots, \frac{n}{2} - 1\}$, we conclude that for any $c > 0$,

$$\mathbf{Pr}\left[\frac{1}{|S_l|}\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i \geq 2\Phi_{\mathrm{c}}(K)^2 - o_n(1), \ \ \forall l \in \left\{0, \ldots, \frac{n}{2} - 1\right\}\right] \geq 1 - O(n^{-c}).$$

Thus, out of all the pairs $\mathbf{X}'_{ij}$ with $j \not\sim i$, with probability at least $1 - O(n^{-c})$ for any $c > 0$, we have that at least a fraction $2\Phi_{\mathrm{c}}(K)^2$ of the pairs are misclassified by the attention mechanism. This concludes part 1 of the theorem.

For part 2, note that by the additive Chernoff bound A.5 we have for any $t \in (0,1)$,

$$\mathbf{Pr}\left[\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i \geq 2|S_l|\Phi_{\mathrm{c}}(K)^2 - |S_l|t\right] \geq 1 - \exp(-|S_l|t^2/4).$$

Since $|S_l| = \frac{nq}{2}(1 \pm o_n(1))$ with probability at least $1/\mathrm{poly}(n)$, we choose $t = \sqrt{\frac{\kappa\Phi_{\mathrm{c}}(K)^2 \log^2 n}{nq}}$ to obtain

$$\mathbf{Pr}\left[\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i \geq nq\Phi_{\mathrm{c}}(K)^2(1 \pm o_n(1)) - \sqrt{\kappa nq\Phi_{\mathrm{c}}(K)^2 \log^2 n}\right] \geq 1 - O(n^{-\frac{\kappa}{4}\Phi_{\mathrm{c}}(K)^2 \log n}).$$

Now note that if $q > \frac{\kappa \log^2 n}{n\Phi_{\mathrm{c}}(K)^2}$ then we have $nq\Phi_{\mathrm{c}}(K)^2 > \kappa \log^2 n$, which implies that

$$nq\Phi_{\mathrm{c}}(K)^2 - \sqrt{\kappa nq\Phi_{\mathrm{c}}(K)^2 \log^2 n} > 0.$$

Hence, in this regime of q,

$$\mathbf{Pr}\left[\sum_{i \in C_0 \cap N_{m(i,l)}} \beta_i > 0\right] \geq 1 - O(n^{-\frac{\kappa}{4}\Phi_{\mathrm{c}}(K)^2 \log n}),$$

and the proof is complete. ■

A.5 Proof of Theorem 6

We first state the formal version of the theorem.

Theorem 10 (Formal restatement of Theorem 6)**.** *Assume that $\|\boldsymbol{\mu}\|_2 \leq K\sigma$ and $\sigma \leq K'$ for some constants K and K'. Moreover, assume that the parameters $(\boldsymbol{w}, \boldsymbol{a}, b) \in \mathbb{R}^d \times \mathbb{R}^2 \times \mathbb{R}$ are bounded by a constant. Then, with probability at least $1 - o_n(1)$ over the data $(\mathbf{X}, \mathbf{A}) \sim$ CSBM$(n, p, q, \boldsymbol{\mu}, \sigma^2)$, there exists a subset $\mathcal{A} \subseteq [n]$ with cardinality at least $n - o(\sqrt{n})$ such that for all $i \in \mathcal{A}$ the following hold:*

1. *There is a subset $J_{i,0} \subseteq N_i \cap C_0$ with cardinality at least $\frac{9}{10}|N_i \cap C_0|$, such that $\gamma_{ij} = \Theta(1/|N_i|)$ for all $j \in J_{i,0}$.*
2. *There is a subset $J_{i,1} \subseteq N_i \cap C_1$ with cardinality at least $\frac{9}{10}|N_i \cap C_1|$, such that $\gamma_{ij} = \Theta(1/|N_i|)$ for all $j \in J_{i,1}$.*

For $i \in [n]$ let $\boldsymbol{g}_i$ be independent Gaussian random variables with mean 0 and variance 1, so we have $\mathbf{X}_i = -\boldsymbol{\mu} + \sigma\boldsymbol{g}_i$ for $i \in C_0$ and $\mathbf{X}_i = \boldsymbol{\mu} + \sigma\boldsymbol{g}_i$ for $i \in C_1$. Moreover, since the parameters $(\boldsymbol{w}, \boldsymbol{a}, b) \in \mathbb{R}^d \times \mathbb{R}^2 \times \mathbb{R}$ are bounded, we can write $\boldsymbol{w} = R\hat{\boldsymbol{w}}$ and $\boldsymbol{a} = R'\hat{\boldsymbol{a}}$ where $\hat{\boldsymbol{w}}$ and $\hat{\boldsymbol{a}}$ are unit vectors and R and R' are some constants. We define the following sets which will become useful later in our computation of γ_{ij}'s.

Define

$$\mathcal{A} \stackrel{\text{def}}{=} \left\{ i \in [n] \;\middle|\; \begin{array}{l} |\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i| \leq 10\sqrt{\log(n(p+q))}, \text{ and} \\ |\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j| \leq 10\sqrt{\log(n(p+q))}, \ \forall j \in N_i \end{array} \right\}.$$

For $i \in [n]$ define

$$\begin{aligned}
J_{i,0} &\stackrel{\text{def}}{=} \left\{ j \in N_i \cap C_0 \mid |\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j| \leq \sqrt{10} \right\}, \\
J_{i,1} &\stackrel{\text{def}}{=} \left\{ j \in N_i \cap C_1 \mid |\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j| \leq \sqrt{10} \right\}, \\
B_{i,0}^t &\stackrel{\text{def}}{=} \left\{ j \in N_i \cap C_0 \mid 2^{t-1} \leq \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j \leq 2^t \right\}, \ t = 1, 2, \ldots, T, \\
B_{i,1}^t &\stackrel{\text{def}}{=} \left\{ j \in N_i \cap C_1 \mid 2^{t-1} \leq \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j \leq 2^t \right\}, \ t = 1, 2, \ldots, T,
\end{aligned}$$

where $T \stackrel{\text{def}}{=} \left\lceil \log_2 \left(10\sqrt{\log(n(p+q))} \right) \right\rceil$.

We start with a few claims about the sizes of these sets.

Claim A.16. *With probability at least* $1 - o(1)$*, we have that* $|\mathcal{A}| \geq n - o(\sqrt{n})$*.*

Proof: Because $|\hat{\boldsymbol{a}}_2| \leq 1$ we know that $\mathcal{A}$ is a superset of $\mathcal{A}'$ where

$$\mathcal{A}' \overset{\text{def}}{=} \left\{ i \in [n] \;\middle|\; \begin{array}{l} |\hat{\boldsymbol{w}}^T \boldsymbol{g}_i| \leq 10\sqrt{\log(n(p+q))}, \text{ and} \\ |\hat{\boldsymbol{w}}^T \boldsymbol{g}_j| \leq 10\sqrt{\log(n(p+q))}, \ \forall j \in N_i \end{array} \right\}.$$

We give a lower bound for $|\mathcal{A}'|$. Denote $b \overset{\text{def}}{=} \mathbf{Pr}\left(|\hat{\boldsymbol{w}}^T \boldsymbol{g}_i| \geq 10\sqrt{\log(n(p+q))}\right)$. Then

$$\frac{1}{2}\frac{1}{10\sqrt{\log(n(p+q))}} e^{-50\log(n(p+q))} \leq b \leq \frac{1}{10\sqrt{\log(n(p+q))}} e^{-50\log(n(p+q))}$$

Apply the multiplicative Chernoff bound

$$\mathbf{Pr}\left[\sum_{i\in[n]} \mathbf{1}_{\left\{|\hat{\boldsymbol{w}}^T \boldsymbol{g}_i| \geq 10\sqrt{\log(n(p+q))}\right\}} \geq nb(1+\delta)\right] \leq e^{-\frac{1}{3}nb\delta^2}$$

and set $\delta = \frac{1}{\sqrt{n}}(10\sqrt{\log(n(p+q))})(n(p+q))^{25}$, we see that with probability at least $1 - o(1)$,

$$\left|\left\{ i \in [n] \mid |\hat{\boldsymbol{w}}^T \boldsymbol{g}_i| \geq 10\sqrt{\log(n(p+q))} \right\}\right| \leq \frac{\sqrt{n}}{(n(p+q))^{25}}.$$

This means that

$$\begin{aligned} &\left|\left\{ i \in [n] \mid |\hat{\boldsymbol{w}}^T \boldsymbol{g}_i| \geq 10\sqrt{\log(n(p+q))} \text{ or } \exists j \in N_i \text{ such that } |\hat{\boldsymbol{w}}^T \boldsymbol{g}_j| \geq 10\sqrt{\log(n(p+q))} \right\}\right| \\ \leq\ & \frac{\sqrt{n}}{(n(p+q))^{25}} \cdot \frac{n}{2}(p+q)(1 \pm o(1)) = \frac{\sqrt{n}}{2(n(p+q))^{24}}(1 \pm o(1)) = o(\sqrt{n}). \end{aligned}$$

Therefore we have

$$|\mathcal{A}'| \geq n - o(\sqrt{n}).$$

■

Claim A.17. *With probability at least* $1 - o(1)$*, we have that for all* $i \in [n]$*,*

$$|J_{i,0}| \geq \frac{9}{10}|N_i \cap C_0| \ \text{ and } \ |J_{i,1}| \geq \frac{9}{10}|N_i \cap C_1|.$$

Proof: We prove the result for $J_{i,0}$, the result for $J_{i,1}$ follows analogously. First fix $i \in [n]$. For each $j \in |N_i \cap C_0|$ we have that

$$\mathbf{Pr}[|\hat{\boldsymbol{a}}_2 \boldsymbol{w}^T \boldsymbol{g}_j| \geq \sqrt{10}] \leq \mathbf{Pr}[|\boldsymbol{w}^T \boldsymbol{g}_j| \geq \sqrt{10}] \leq e^{-50}.$$

Denote $J_{i,0}^c \overset{\text{def}}{=} (N_i \cap C_0) \setminus J_{i,0}$. We have that

$$\mathbf{E}[|J_{i,0}^c|] = \mathbf{E}\left[\sum_{j \in N_i \cap C_0} \mathbf{1}_{\left\{|\hat{\boldsymbol{a}}_2 \boldsymbol{w}^T \boldsymbol{g}_j| \geq \sqrt{10}\right\}}\right] \leq e^{-50}|N_i \cap C_0|,$$

Apply Chernoff's inequality (Theorem 2.3.4 in Vershynin (2018)) we have

$$\begin{aligned} \mathbf{Pr}\left[|J_{i,0}^c| \geq \frac{1}{10}|N_i \cap C_0|\right] &\leq e^{-\mathbf{E}[|J_{i,0}^c|]}\left(\frac{e\,\mathbf{E}[|J_{i,0}^c|]}{|N_i \cap C_0|/10}\right)^{|N_i \cap C_0|/10} \\ &\leq \left(\frac{ee^{-50}|N_i \cap C_0|}{|N_i \cap C_0|/10}\right)^{|N_i \cap C_0|/10} \\ &= \exp\left(-\left(\frac{1}{2} - \frac{\log 10}{10} - \frac{1}{10}\right)|N_i \cap C_0|\right) \\ &\leq \exp\left(-\frac{4}{25}|N_i \cap C_0|\right). \end{aligned}$$

Apply the union bound we get

$$\mathbf{Pr}\left[|J_{i,0}| \geq \frac{9}{10}|C_0 \cap N_i|, \forall i \in [n]\right] \geq 1 - \sum_{i\in[n]} e^{-\frac{4}{25}|N_i\cap C_0|} \geq (1-o(1))\left(1 - ne^{-\frac{2n\min\{p,q\}(1\pm o(1))}{25}}\right) = 1-o(1).$$

The second last inequality follows because $|N_i \cap C_0| \geq \frac{n}{2}\min\{p,q\}(1 \pm o(1)) = \frac{nq}{2}(1 \pm o(1))$ under degree concentration for all $i \in [n]$. Moreover, since we have used the degree concentration, this introduces the additional multiplicative $(1-o(1))$ term in the probability lower bound. The last equality is due to our assumption that $p, q = \Omega(\frac{\log^2 n}{n})$. ■

Claim A.18. *With probability at least* $1-o(1)$*, we have that for all* $i \in [n]$ *and for all* $t \in [T]$*,*

$$|B_{i,0}^t| \leq \mathbf{E}[|B_{i,0}^t|] + \sqrt{T}|N_i \cap C_0|^{\frac{4}{5}} \;\; \textit{and} \;\; |B_{i,1}^t| \leq \mathbf{E}[|B_{i,1}^t|] + \sqrt{T}|N_i \cap C_1|^{\frac{4}{5}}.$$

Proof: We prove the result for $B_{i,0}^t$, and the result for $B_{i,1}^t$ follows analogously. First fix $i \in [n]$ and $t \in [T]$. By the additive Chernoff inequality we have

$$\mathbf{Pr}\left(|B_{i,0}^t| \geq \mathbf{E}[|B_{i,0}^t|] + |N_i \cap C_0| \cdot \sqrt{T}|N_i \cap C_0|^{-\frac{1}{5}}\right) \leq e^{-2T|N_i\cap C_0|^{3/5}}.$$

Taking a union bound over all $i \in [n]$ and $t \in [T]$ we get

$$\begin{aligned}&\mathbf{Pr}\left[\bigcup_{i\in[n]}\bigcup_{t\in[T]}\left\{|B_{i,0}^t| \geq \mathbf{E}[|B_{i,0}^t|] + \sqrt{T}|N_i \cap C_0|^{\frac{4}{5}}\right\}\right] \\ &\leq \; nT\exp\left(-2T\left(\frac{n}{2}\min\{p,q\}(1\pm o(1))\right)^{3/5}\right) + o(1) \; = \; o(1),\end{aligned}$$

where the last equality follows from Assumption 1 that $p, q = \Omega(\frac{\log^2 n}{n})$, and hence

$$nT\exp\left(-2T\left(\frac{n}{2}\min\{p,q\}(1\pm o(1))\right)^{3/5}\right) = nT\exp\left(-\omega\left(\sqrt{2}T\log n\right)\right) = O\left(\frac{1}{n^c}\right)$$

for some absolute constant $c > 0$. Moreover, we have used degree concentration, which introduced the additional additive $o(1)$ term in the probability upper bound. Therefore we have

$$\mathbf{Pr}\left[|B_{i,0}^t| \leq \mathbf{E}[|B_{i,0}^t|] + \sqrt{T}|N_i \cap C_0|^{\frac{4}{5}}, \forall i \in [n] \; \forall t \in [T]\right] \geq 1 - o(1).$$

■

Proof of Theorem 10: We start by defining an event $\boldsymbol{\mathcal{E}}^*$ which is the intersection of the following events over the randomness of $\mathbf{A}$ and $\{\epsilon_i\}_i$ and $\mathbf{X}_i = (2\epsilon_i - 1)\boldsymbol{\mu} + \sigma\boldsymbol{g}_i$,

- $\boldsymbol{\mathcal{E}}_0$ is the event that for each $i \in [n]$, $|C_0 \cap N_i| = \frac{n}{2}((1-\epsilon_i)p + \epsilon_i q)(1 \pm o(1))$ and $|C_1 \cap N_i| = \frac{n}{2}((1-\epsilon_i)q + \epsilon_i p)(1 \pm o(1))$.
- $\boldsymbol{\mathcal{E}}_1$ is the event that $|\mathcal{A}| \geq n - o(\sqrt{n})$.
- $\boldsymbol{\mathcal{E}}_2$ is the event that $|J_{i,0}| \geq \frac{9}{10}|N_i \cap C_0|$ and $|J_{i,1}| \geq \frac{9}{10}|N_i \cap C_1|$ for all $i \in [n]$.
- $\boldsymbol{\mathcal{E}}_3$ is the event that $|B_{i,0}^t| \leq \mathbf{E}[|B_{i,0}^t|] + \sqrt{T}|N_i \cap C_0|^{\frac{4}{5}}$ and $|B_{i,1}^t| \leq \mathbf{E}[|B_{i,1}^t|] + \sqrt{T}|N_i \cap C_1|^{\frac{4}{5}}$ for all $i \in [n]$ and for all $t \in [T]$.

By Claims A.16, A.17, A.18, we get that with probability at least $1-o(1)$, the event $\boldsymbol{\mathcal{E}}^* = \bigcap_{i=0}^{3}\boldsymbol{\mathcal{E}}_i$ holds. We will show that under event $\boldsymbol{\mathcal{E}}^*$, for all $i \in \mathcal{A}$, for all $j \in J_{i,c}$ where $c \in \{0,1\}$, we have $\gamma_{ij} = \Theta(1/|N_i|)$. This will prove Theorem 10.

Fix $i \in \mathcal{A}$ and some $j \in J_{i,0}$. Let us consider

$$\begin{aligned}
\gamma_{ij} &= \frac{\exp\big(\mathrm{LeakyRelu}(\boldsymbol{a}_1 \boldsymbol{w}^T \mathbf{X}_i + \boldsymbol{a}_2 \boldsymbol{w}^T \mathbf{X}_j + b)\big)}{\sum_{k \in N_i} \exp\left(\mathrm{LeakyRelu}(\boldsymbol{a}_1 \boldsymbol{w}^T \mathbf{X}_i + \boldsymbol{a}_2 \boldsymbol{w}^T \mathbf{X}_k + b)\right)} \\
&= \frac{\exp\big(\sigma R R' \ \mathrm{LeakyRelu}(\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b')\big)}{\sum_{k \in N_i} \exp\left(\sigma R R' \ \mathrm{LeakyRelu}(\kappa_{ik} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k + b')\right)} \\
&= \frac{1}{\sum_{k \in N_i} \exp(\Delta_{ik} - \Delta_{ij})}
\end{aligned}$$

where for $l \in N_i$, we denote

$$\begin{aligned}
\kappa_{il} &\overset{\text{def}}{=} (2\epsilon_i - 1)\hat{\boldsymbol{w}}^T \boldsymbol{\mu}/\sigma + (2\epsilon_l - 1)\hat{\boldsymbol{w}}^T \boldsymbol{\mu}/\sigma, \\
\Delta_{il} &\overset{\text{def}}{=} \sigma R R' \ \mathrm{LeakyRelu}(\kappa_{il} + \hat{\boldsymbol{a}}_1 \boldsymbol{w}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \boldsymbol{w}^T \boldsymbol{g}_l + b'),
\end{aligned}$$

and $b = \sigma R R' b'$. We will show that $\sum_{k \in N_i} \exp(\Delta_{ik} - \Delta_{ij}) = \Theta(|N_i|)$ and hence conclude that $\gamma_{ij} = \Theta(1/|N_i|)$.

First of all, note that since $\|\boldsymbol{\mu}\|_2 \leq K\sigma$ for some absolute constant K, we know that

$$|\kappa_{il}| \leq \sqrt{2}K = O(1).$$

Let us assume that $\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i \geq 0$ and consider the following two cases regarding the magnitude of $\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i$.

<u>Case 1.</u> If $\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b' < 0$, then

$$\begin{aligned}
\Delta_{ik} - \Delta_{ij} &= \sigma R R' \Big(\mathrm{LeakyRelu}(\kappa_{ik} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k + b') \\
&\qquad - \mathrm{LeakyRelu}(\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b') \Big) \\
&= \sigma R R' \Big(\mathrm{LeakyRelu}(\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k \pm O(1)) \\
&\qquad - \beta(\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b') \Big) \\
&= \sigma R R' \left(\mathrm{LeakyRelu}(\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k \pm O(1)) \pm O(1) \right) \\
&= \sigma R R' \left(\Theta(\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k) \pm O(1) \right),
\end{aligned}$$

where β is the slope of $\mathrm{LeakyRelu}(x)$ for $x < 0$. Here, the second equality follows from $|\kappa_{ik} + b'| \leq \sqrt{2}K + |b'| = O(1)$ and $\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b' < 0$. The third equality follows from

- We have $j \in J_{i,0}$ and hence $|\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j| = O(1)$;
- We have $\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b' < 0$, so $\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i < |\kappa_{ij}| + |\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j| + |b'| = O(1)$, moreover, because $\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i \geq 0$, we get that $|\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i| = O(1)$;
- We have $|\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b'| \leq |\hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i| + |\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j| + |\kappa_{ij} + b'| = O(1) + O(1) + O(1) = O(1)$.

<u>Case 2.</u> If $\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b' \geq 0$, then

$$\begin{aligned}
\Delta_{ik} - \Delta_{ij} &= \sigma R R' \Big(\mathrm{LeakyRelu}(\kappa_{ik} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k + b') \\
&\qquad - \mathrm{LeakyRelu}(\kappa_{ij} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j + b') \Big) \\
&= \sigma R R' \Big(\mathrm{LeakyRelu}(\kappa_{ik} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k + b') \\
&\qquad - \kappa_{ij} - \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i - \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_j - b' \Big) \\
&= \sigma R R' \left(\mathrm{LeakyRelu}(\kappa_{ik} + \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i + \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k + b') - \hat{\boldsymbol{a}}_1 \hat{\boldsymbol{w}}^T \boldsymbol{g}_i \pm O(1) \right) \\
&\begin{cases} = \sigma R R' \left(\Theta(\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k) \pm O(1) \right), & \text{if } k \in J_{i,0} \cup J_{i,1} \\ \leq \sigma R R' \left(O(\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k) \pm O(1) \right), & \text{otherwise.} \end{cases}
\end{aligned}$$

To see the last (in)equality in the above, consider the following cases:

1. If $k \in J_{i,0} \cup J_{i,1}$, then there are two cases depending on the sign of $\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b'$.
   - If $\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b' \geq 0$, then we have that
   $$\begin{aligned} & \text{LeakyRelu}(\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b') - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b' - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + \kappa_{ik} + b' \pm O(1) \\ = \; & \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k \pm O(1). \end{aligned}$$
   - If $\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b' < 0$, then because $\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \geq 0$ and $|\kappa_{ik} + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b'| \leq |\kappa_{ik}| + |\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k| + |b'| = O(1)$, we know that $\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i < |\kappa_{ik}| + |\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k| + |b'| = O(1)$ and $|\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b'| = O(1)$. Therefore it follows that
   $$\begin{aligned} & \text{LeakyRelu}(\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b') - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \text{LeakyRelu}(\pm O(1)) - O(1) \pm O(1) \\ = \; & \pm O(1) \\ = \; & \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k \pm O(1) \end{aligned}$$
   where the last equality is due to the fact that $k \in J_{i,0} \cup J_{i,1}$ so $|\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k| = O(1)$.
2. If $k \notin J_{i,0} \cup J_{i,1}$, then there are two cases depending on the sign of $\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b'$.
   - If $\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b' \geq 0$, then we have that
   $$\begin{aligned} & \text{LeakyRelu}(\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b') - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b' - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + \kappa_{ik} + b' \pm O(1) \\ = \; & \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k \pm O(1). \end{aligned}$$
   - If $\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b' < 0$, then we have that,
   $$\begin{aligned} & \text{LeakyRelu}(\kappa_{ik} + \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + b') - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \beta\kappa_{ik} + \beta\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i + \beta\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k + \beta b' - \boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ = \; & \beta\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k - (1-\beta)\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \pm O(1) \\ \leq \; & \beta\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k \pm O(1), \end{aligned}$$
   where β is the slope of $\text{LeakyRelu}(\cdot)$.

Combining the two cases regarding the magnitude of $\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i$ and our assumption that $\sigma, R, R = O(1)$, so far we have showed that, for any i such that $\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i \geq 0$, for all $j \in J_{i,0}$, we have

$$\Delta_{ik} - \Delta_{ij} = \begin{cases} \Theta(\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k) \pm O(1), & \text{if } k \in J_{i,0} \cup J_{i,1} \\ O(\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k) \pm O(1), & \text{otherwise.} \end{cases} \tag{19}$$

By following a similar argument, one can show that Equation 19 holds for any i such that $\boldsymbol{\hat{a}}_1 \boldsymbol{\hat{w}}^T \boldsymbol{g}_i < 0$.

Let us now compute

$$\sum_{k \in N_i} \exp(\Delta_{ik} - \Delta_{ij}) = \sum_{k \in N_i \cap C_0} \exp(\Delta_{ik} - \Delta_{ij}) + \sum_{k \in N_i \cap C_1} \exp(\Delta_{ik} - \Delta_{ij})$$

for some $j \in J_{i,0}$. Let us focus on $\sum_{k \in N_i \cap C_0} \exp(\Delta_{ik} - \Delta_{ij})$ first. We will show that $\Omega(|N_i \cap C_0|) \leq \sum_{k \in N_i \cap C_0} \exp(\Delta_{ik} - \Delta_{ij}) \leq O(|N_i|)$.

First of all, we have that

$$\begin{aligned} \sum_{k \in N_i \cap C_0} \exp(\Delta_{ik} - \Delta_{ij}) \geq \sum_{k \in J_{i,0}} \exp(\Delta_{ik} - \Delta_{ij}) &= \sum_{k \in J_{i,0}} \exp\left(\Theta(\boldsymbol{\hat{a}}_2 \boldsymbol{\hat{w}}^T \boldsymbol{g}_k) \pm O(1)\right) \\ \geq \sum_{k \in J_{i,0}} e^{c_1} = |J_{i,0}| e^{c_1} &= \Omega(|N_i \cap C_0|), \end{aligned} \tag{20}$$

where c_1 is an absolute constant (possibly negative). On the other hand, consider the following partition of $N_i \cap C_0$:

$$\begin{aligned} P_1 &\overset{\text{def}}{=} \{k \in N_i \cap C_0 \mid \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k \leq 1\}, \\ P_2 &\overset{\text{def}}{=} \{k \in N_i \cap C_0 \mid \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k \geq 1\}. \end{aligned}$$

It is easy to see that

$$\sum_{k \in P_1} \exp(\Delta_{ik} - \Delta_{ij}) \leq \sum_{k \in P_1} \exp\left(O(\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k) \pm O(1)\right) \leq \sum_{k \in P_1} e^{c_2} = |P_1| e^{c_2} = O(|N_i \cap C_0|), \tag{21}$$

where c_2 is an absolute constant. Moreover, because $i \in \mathcal{A}$ we have that $P_2 \subseteq \bigcup_{t \in [T]} B_{i,0}^t$. It follows that

$$\begin{aligned} \sum_{k \in P_2} \exp(\Delta_{ik} - \Delta_{ij}) &= \sum_{t \in [T]} \sum_{k \in B_{i,0}^t} \exp(\Delta_{ik} - \Delta_{ij}) \\ &\leq \sum_{t \in [T]} \sum_{k \in B_{i,0}^t} \exp\left(O(\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k) \pm O(1)\right) \\ &\leq \sum_{t \in [T]} |B_{i,0}^t| e^{c_3 2^t}, \end{aligned} \tag{22}$$

where c_3 is an absolute constant. We can upper bound the above quantity as follows. Under the Event $\boldsymbol{\mathcal{E}}^*$, we have that

$$|B_{i,0}^t| \leq m_t + \sqrt{T} |N_i \cap C_0|^{\frac{4}{5}}, \text{ for all } t \in [T],$$

where

$$\begin{aligned} m_t \overset{\text{def}}{=} \mathbf{E}[|B_{i,0}^t|] &= \sum_{k \in N_i \cap C_0} \mathbf{Pr}(2^{t-1} \leq \hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k \leq 2^t) \leq \sum_{k \in N_i \cap C_0} \mathbf{Pr}[\hat{\boldsymbol{a}}_2 \hat{\boldsymbol{w}}^T \boldsymbol{g}_k \geq 2^{t-1}] \\ &\leq \sum_{k \in N_i \cap C_0} \mathbf{Pr}[\hat{\boldsymbol{w}}^T \boldsymbol{g}_k \geq 2^{t-1}] \leq |N_i \cap C_0| e^{-2^{2t-3}}. \end{aligned}$$

It follows that

$$\begin{aligned} \sum_{t \in [T]} |B_{i,0}^t| e^{c_3 2^t} &\leq \sum_{t \in [T]} \left(|N_i \cap C_0| e^{-2^{2t-3}} + \sqrt{T} |N_i \cap C_0|^{\frac{4}{5}}\right) e^{c_3 2^t} \\ &\leq |N_i \cap C_0| \sum_{t=1}^{\infty} e^{-2^{2t-3}} e^{c_3 2^t} + \sum_{t \in [T]} \sqrt{T} |N_i \cap C_0|^{\frac{4}{5}} e^{c_3 2^T} \\ &\leq c_4 |N_i \cap C_0| + o(|N_i|) \\ &\leq O(|N_i|), \end{aligned} \tag{23}$$

where c_4 is an absolute constant. The third inequality in the above follows from

- The series $\sum_{t=1}^{\infty} e^{-2^{2t-3}} e^{c_3 2^t}$ converges absolutely for any constant c_3;
- The sum $\sum_{t \in [T]} \sqrt{T} |N_i \cap C_0|^{\frac{4}{5}} e^{c_3 2^T} = T^{\frac{3}{2}} |N_i \cap C_0|^{\frac{4}{5}} e^{c_3 2^T} = o(|N_i|)$ because

$$\begin{aligned} \log\left(T^{\frac{3}{2}} e^{c_3 2^T}\right) &= \frac{3}{2} \log \left\lceil \log_2 \left(10\sqrt{\log(n(p+q))}\right)\right\rceil + c_3 2^{\left\lceil \log_2 \left(10\sqrt{\log(n(p+q))}\right)\right\rceil} \\ &\leq \frac{3}{2} \log \left\lceil \log_2 \left(10\sqrt{\log(n(p+q))}\right)\right\rceil + 20 c_3 \sqrt{\log(n(p+q))} \\ &\leq O\left(\frac{1}{c} \log(n(p+q))\right), \end{aligned}$$

for any $c > 0$. In particular, by picking $c > 5$ we see that $T^{\frac{3}{2}} e^{c_3 2^T} \leq O((n(p+q))^{\frac{1}{c}}) \leq o(|N_i|^{\frac{1}{5}})$, and hence we get $T^{\frac{3}{2}} e^{c_3 2^T} |N_i \cap C_0|^{\frac{4}{5}} \leq |N_i|^{\frac{4}{5}} \cdot o(|N_i|^{\frac{1}{5}}) = o(|N_i|)$.

Combining Equations 22 and 23 we get

$$\sum_{k\in P_2} \exp(\Delta_{ik} - \Delta_{ij}) \leq O(|N_i|), \tag{24}$$

and combining Equations 21 and 24 we get

$$\sum_{k\in N_i\cap C_0} \exp(\Delta_{ik} - \Delta_{ij}) = \sum_{k\in P_1} \exp(\Delta_{ik} - \Delta_{ij}) + \sum_{k\in P_1} \exp(\Delta_{ik} - \Delta_{ij}) \leq O(|N_i|). \tag{25}$$

Now, by Equations 20 and 25 we get

$$\Omega(|N_i \cap C_0|) \leq \sum_{k\in N_i\cap C_0} \exp(\Delta_{ik} - \Delta_{ij}) \leq O(|N_i|). \tag{26}$$

It turns out that repeating the same argument for $\sum_{k\in N_i\cap C_1} \exp(\Delta_{ik} - \Delta_{ij})$ yields

$$\Omega(|N_i \cap C_1|) \leq \sum_{k\in N_i\cap C_1} \exp(\Delta_{ik} - \Delta_{ij}) \leq O(|N_i|). \tag{27}$$

Finally, Equations 26 and 27 give us

$$\sum_{k\in N_i} \exp(\Delta_{ik} - \Delta_{ij}) = \Theta(|N_i|),$$

which readily implies

$$\gamma_{ij} = \frac{1}{\sum_{k\in N_i} \exp(\Delta_{ik} - \Delta_{ij})} = \Theta(1/|N_i|)$$

as required. We have showed that for all $i \in \mathcal{A}$ and for all $j \in J_{i,0}$, $\gamma_{ij} = \Theta(1/|N_i|)$. Repeating the same argument we get that the same result holds for all $i \in \mathcal{A}$ and for all $j \in J_{i,1}$, too. Hence, by Claims A.16 and A.17 about the cardinalities of $\mathcal{A}$, $J_{i,0}$ and $J_{i,1}$ we have thus proved Theorem 10. ■

B Additional experimental results

B.1 Ansatz for GAT, MLP-GAT and GCN

For the original GAT architecture we fixed $\boldsymbol{w} = \boldsymbol{\mu}/\|\boldsymbol{\mu}\|_2$ and defined the first head as $\boldsymbol{a}_1 = \frac{1}{\sqrt{2}}(1,1)$ and $b_1 = -\frac{1}{\sqrt{2}}\boldsymbol{w}^T\boldsymbol{\mu}$; The second head is defined as $\boldsymbol{a}_2 = -\boldsymbol{a}_1$ and $b_2 = -b_1$. We briefly discuss the choice of such ansatz. The parameter $\boldsymbol{w}$ is picked based on the optimal Bayes classifier without a graph, and the attention is set such that the first head maintains pairs in C_1 and the second head maintains pairs in C_0.[7] We will clearly see from the results, this choice of ansatz produces good node classification performance (in the "easy regime", where we vary q we clearly see how those performances degrade since GAT linear attention mechanism is unable to separate inter- from intra-edges). More specifically, one may use the same techniques in the proof of Theorem 1 and Corollaries 2 and 3 to prove the node separability results (in this particular case, the result will *depend on* q in contrast to the result we get for MLP-GAT, where the no dependence of q was needed).

For MLP-GAT we use the following choice of ansatz.

$$\tilde{\boldsymbol{w}} \stackrel{\text{def}}{=} \frac{\boldsymbol{\mu}}{\|\boldsymbol{\mu}\|_2}, \qquad \mathbf{S} \stackrel{\text{def}}{=} \begin{bmatrix} 1 & 1 \\ -1 & -1 \\ 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad \boldsymbol{r} \stackrel{\text{def}}{=} \begin{bmatrix} 1 & 1 & -1 & -1 \end{bmatrix}$$

so that the output of the attention model is defined as

$$\boldsymbol{r} \cdot \text{LeakyRelu}\left(\mathbf{S} \cdot \begin{bmatrix} \tilde{\boldsymbol{w}}^T\mathbf{X}_i \\ \tilde{\boldsymbol{w}}^T\mathbf{X}_j \end{bmatrix}\right).$$

This choice of two layer network allows us to bypass the "XOR problem" (Minsky & Papert, 1969) and separate inter- from intra-edges, which is clearly impossible with linear architecture.

For GCN we used the ansatz from Baranwal et al. (2021), which is also $\boldsymbol{w} = \boldsymbol{\mu}/\|\boldsymbol{\mu}\|_2$.

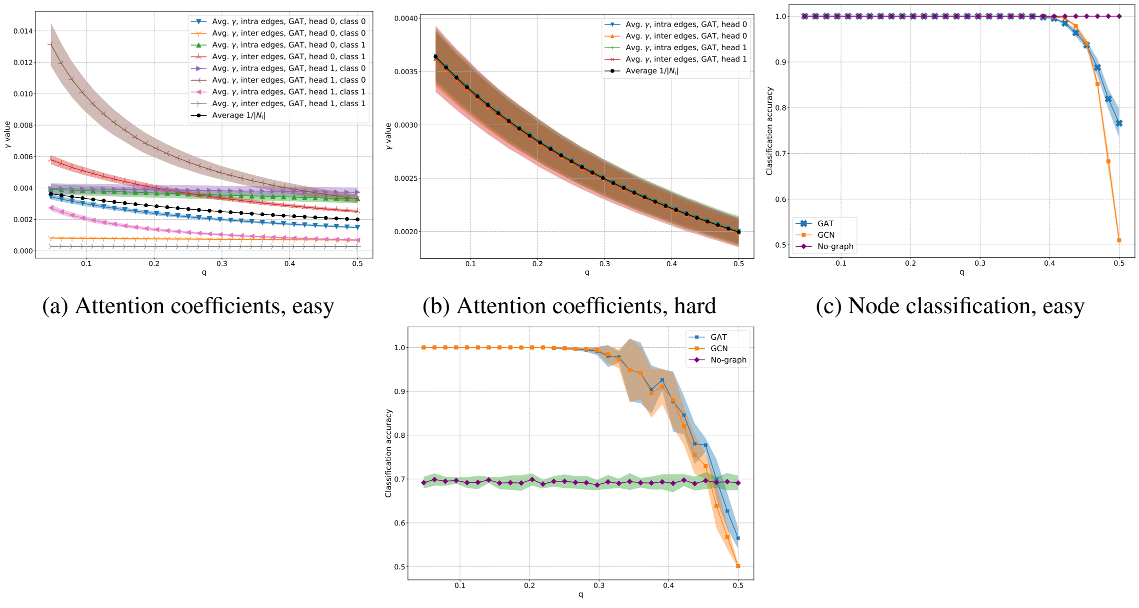
[7] Note that in this case, due to the linearity of the attention mechanism, it will be impossible for the model to keep only γ_{ij} which correspond to intra-class edges.

B.2 Synthetic data

B.2.1 Fixing the distance between the means and varying q

In this subsection, we present additional experimental results for the original GAT (Velickovic et al., 2018) mechanism with two heads, and MLP-GAT on synthetic data. Unless stated otherwise, we use the exact parameter setting as in Subsection 4.1.1.

As explained in the introduction, a GAT head (equipped with a linear attention mechanism) is bound to fail to separate the edges. Recall that when considering the pair space $(\boldsymbol{w}^T\mathbf{X}_i, \boldsymbol{w}^T\mathbf{X}_j)$, we can think of each pair as a two-dimensional Gaussian with means in either one of the four quadrants. For correct edge classification, we need to classify the data originating from the distributions whose means are in the second and fourth quadrant from the data originating from the distributions whose means are in the first and the third quadrants (that is, the problem is a "XOR problem" (Minsky & Papert, 1969)). However, it is readily seen that any linear classifier fails on the above task. In Figure 5a we can clearly see the lack of ability of the heads to ignore inter-edges. For example, head 0 maintains all the intra-edges of class 1 but also maintains the inter-edges between class 1 to class 0. Figure 5b presents the attention coefficients of GAT in the "hard regime" and demonstrates Theorem 6, where most γ concentrate around uniform (GCN) coefficients. In Figure 5c we observe the node classification performance of the GAT model in the "easy regime". As opposed to MLP-GAT, we can clearly see that the node classification performance of GAT is affected by increasing q. Figure 5d demonstrates the node classification performance in the "hard regime" (which is conjectured in Conjecture 7).



(a) Attention coefficients, easy

(b) Attention coefficients, hard

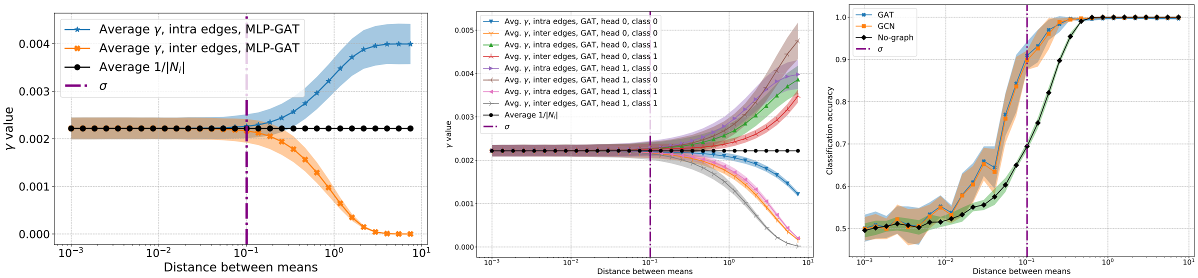
(c) Node classification, easy

(d) Node classification, hard

Figure 5: Attention coefficients and node classification accuracy for GAT.

B.2.2 Fixing q and varying the distance between the means

We consider the case where $q = 0.4$. In Figure 6 we show the attention coefficients for both MLP-GAT and two head GAT as a function of the distance between the means, and the node classification performance of GAT as a function of the distance between the means. In Figure 6a we see that in the "hard regime" MLP-GAT produce attention coefficients that concentrate around uniform (GCN) coefficients, while in the "easy regime" the model is able to maintain only the γ that correspond to intra-class edges (as stated in Corollary 2) while ignoring all coefficients corresponding to inter-class edges. In Figure 6b we observe that in the "hard regime" the attention coefficients of GAT concentrate around the uniform coefficients (as proved in Theorem 6), while in the "easy regime", even though the attention coefficients concentrate, GAT is not able to distinguish intra from inter edges. As explained before, this is due to the fact that a linear attention mechanism is bound to fail on the “XOR problem”. Lastly, in Figure 6c we show classification results for GAT. Note that in the “easy regime” GAT achieves perfect separability. However, as the distance between the means decreases, GAT begins to misclassify nodes.



(a) Attention coeff. of MLP-GAT.

(b) Attention coeff. of GAT.

(c) Node classification accuracy.

Figure 6: Attention coefficients MLP-GAT/GAT and node classification for GAT.

B.3 Real-world data

B.3.1 Ansatz

In Figure 7 we present the results for MLP-GAT on CiteSeer using the chosen ansatz.

In Figure 8 we present the results for MLP-GAT on Cora using the chosen ansatz.

In Figure 9 we present the results for MLP-GAT on PubMed using the chosen ansatz.

In Figure 10 we present the results for GAT on CiteSeer using the chosen ansatz.

In Figure 11 we present the results for GAT on Cora using the chosen ansatz.

In Figure 12 we present the results for GAT on PubMed using the chosen ansatz.

B.3.2 Training

For experiments on real data, we also used PyTorch Geometric (Fey & Lenssen, 2019) to train the models GAT, MLP-GAT, GCN, and the linear classifier. We train the models using the Adam optimizer with a learning rate of 10^{-3}, weight decay 10^{-3}, and 200 epochs. We terminate the process if the average binary cross-entropy loss is less than 10^{-2}. We report the results of the last epoch.

It is important to note that for MLP-GAT the results that we get from the training process are very similar to the ones reported in the main paper. One difference that we observed is that for a very large distance between the means the trained parameters of MLP-GAT resulted in some misclassifications for the edges. This happens because in the “easy regime” the graph is not needed at all and there must exist many parameter settings that achieve a near-perfect node classification without necessarily distinguishing intra-class from inter-class edges.

For GAT we also observe similar trends for training to the ones that we observed using an ansatz. One difference is the behavior of γ, which exhibit some irregular behavior when the distance between the means is very large. Again, this does not seem to affect node classification accuracy. In the case where the distance between the means is small, we do observe that the average γ is close to uniform, which is also what we have proved in the main paper for the CSBM model.

In Figure 13 we present the results for MLP-GAT on CiteSeer under training.

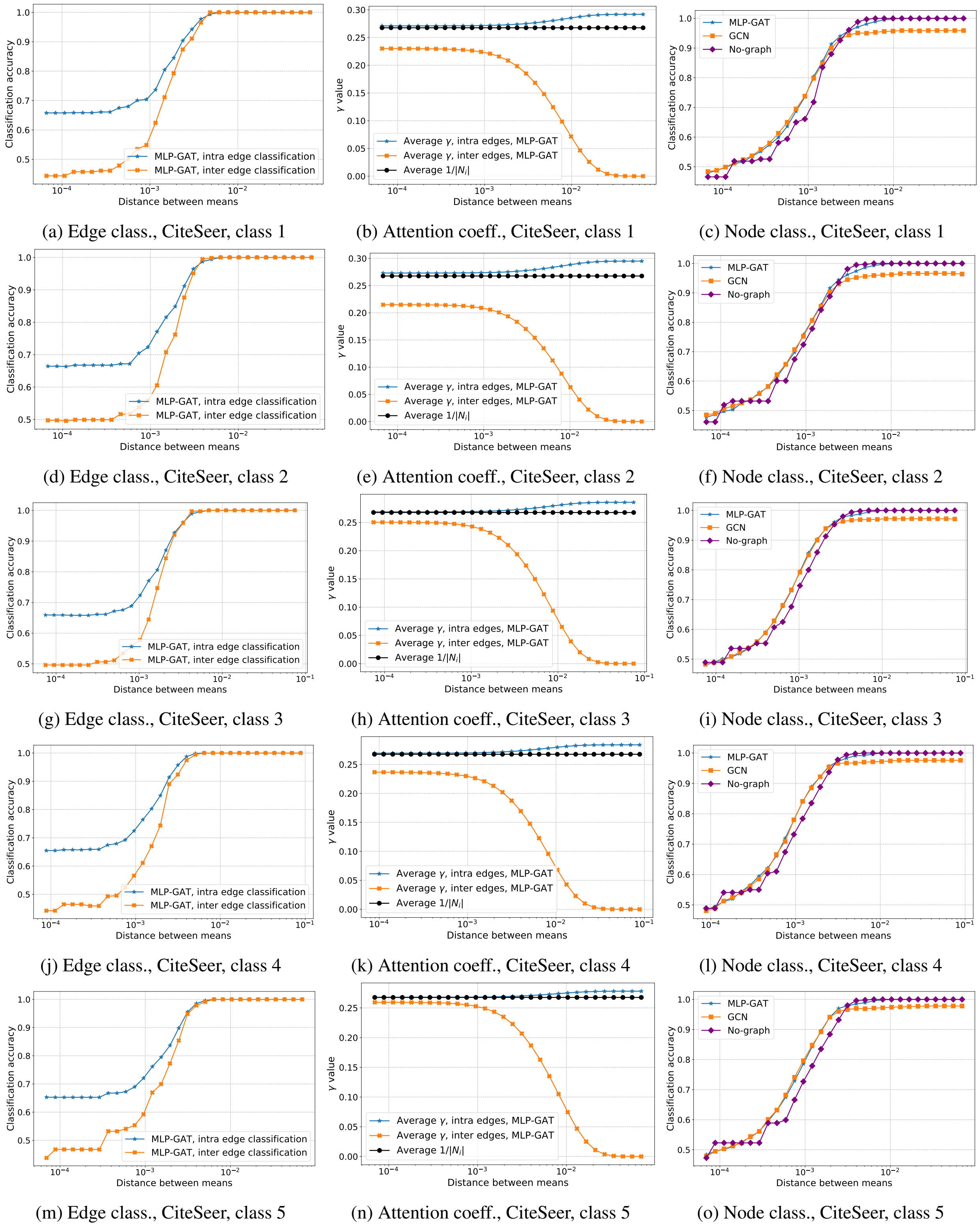
In Figure 14 we present the results for MLP-GAT on Cora under training.

In Figure 15 we present the results for MLP-GAT on PubMed under training.

In Figure 16 we present the results for GAT on CiteSeer under training.

In Figure 17 we present the results for GAT on Cora under training.

In Figure 18 we present the results for GAT on PubMed under training.



(a) Edge class., CiteSeer, class 1

(b) Attention coeff., CiteSeer, class 1

(c) Node class., CiteSeer, class 1

(d) Edge class., CiteSeer, class 2

(e) Attention coeff., CiteSeer, class 2

(f) Node class., CiteSeer, class 2

(g) Edge class., CiteSeer, class 3

(h) Attention coeff., CiteSeer, class 3

(i) Node class., CiteSeer, class 3

(j) Edge class., CiteSeer, class 4

(k) Attention coeff., CiteSeer, class 4

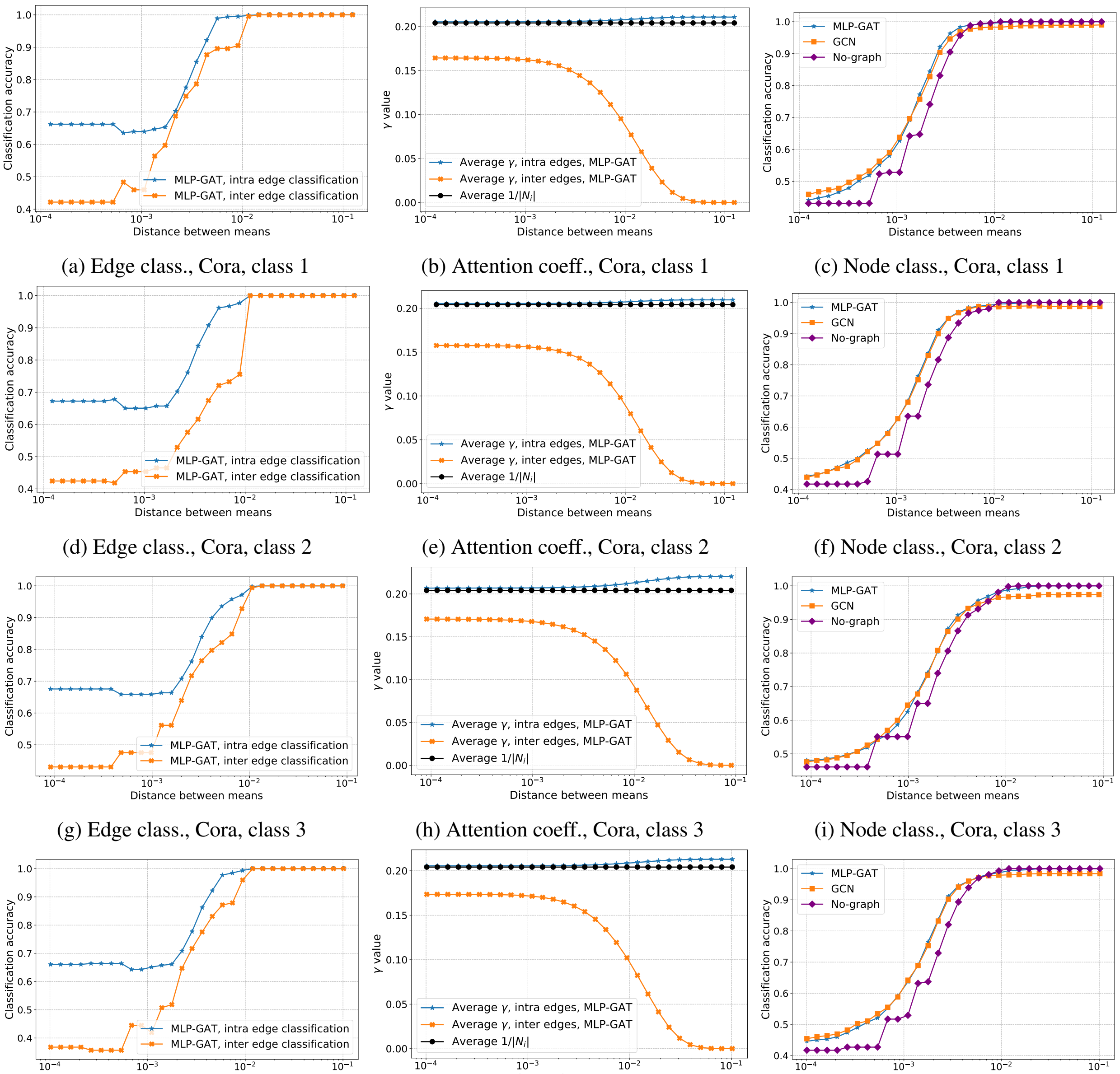
(l) Node class., CiteSeer, class 4

(m) Edge class., CiteSeer, class 5

(n) Attention coeff., CiteSeer, class 5

(o) Node class., CiteSeer, class 5

Figure 7: Ansatz MLP-GAT on CiteSeer.



(a) Edge class., Cora, class 1

(b) Attention coeff., Cora, class 1

(c) Node class., Cora, class 1

(d) Edge class., Cora, class 2

(e) Attention coeff., Cora, class 2

(f) Node class., Cora, class 2

(g) Edge class., Cora, class 3

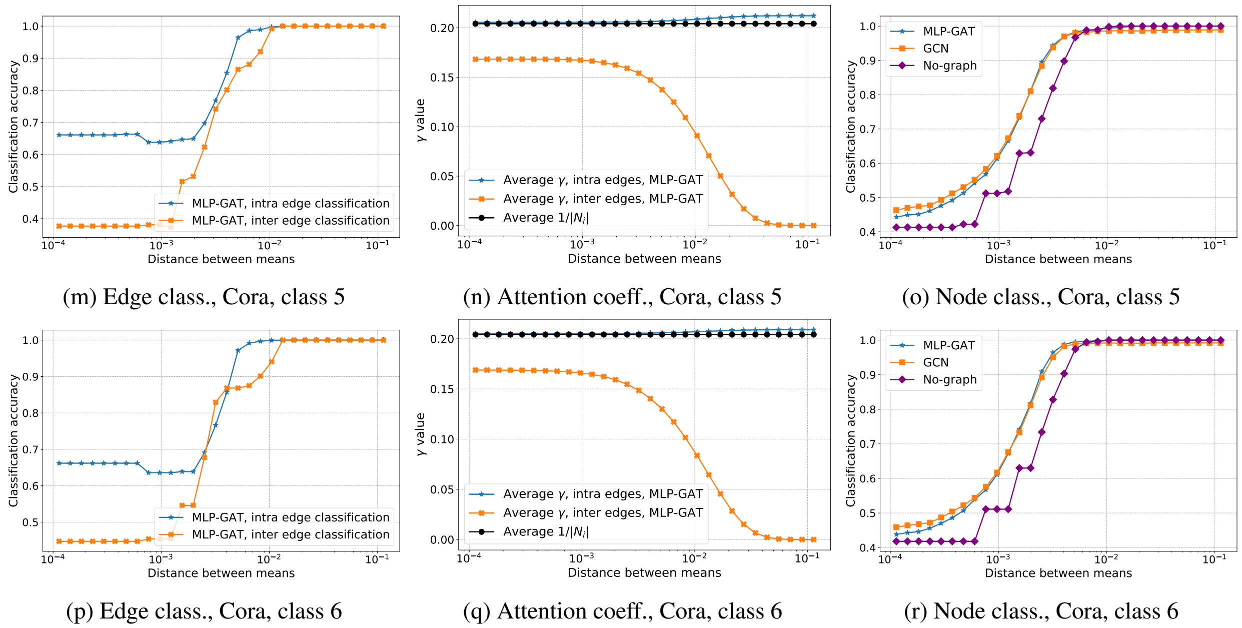
(h) Attention coeff., Cora, class 3

(i) Node class., Cora, class 3

(j) Edge class., Cora, class 4

(k) Attention coeff., Cora, class 4

(l) Node class., Cora, class 4



(m) Edge class., Cora, class 5

(n) Attention coeff., Cora, class 5

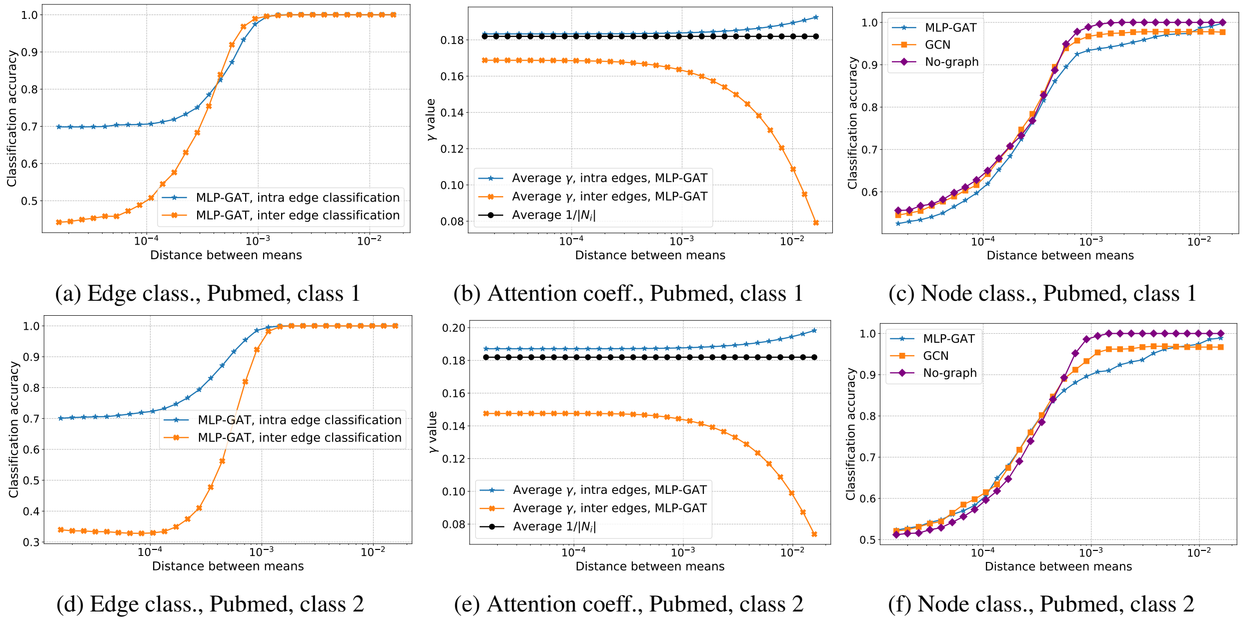
(o) Node class., Cora, class 5

(p) Edge class., Cora, class 6

(q) Attention coeff., Cora, class 6

(r) Node class., Cora, class 6

Figure 8: Ansatz MLP-GAT on Cora.



(a) Edge class., Pubmed, class 1

(b) Attention coeff., Pubmed, class 1

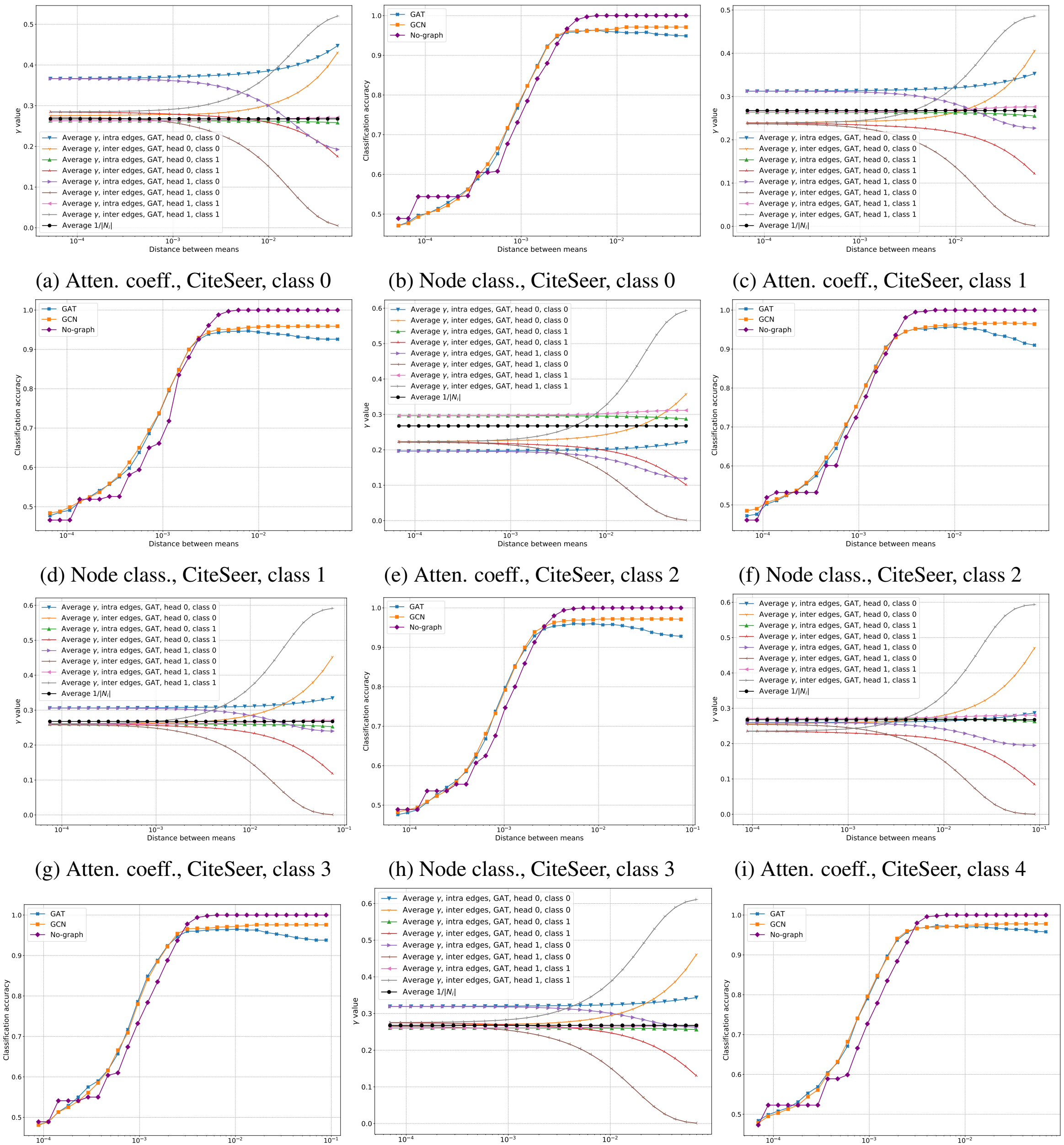
(c) Node class., Pubmed, class 1

(d) Edge class., Pubmed, class 2

(e) Attention coeff., Pubmed, class 2

(f) Node class., Pubmed, class 2

Figure 9: Ansatz MLP-GAT on PubMed.



(a) Atten. coeff., CiteSeer, class 0

(b) Node class., CiteSeer, class 0

(c) Atten. coeff., CiteSeer, class 1

(d) Node class., CiteSeer, class 1

(e) Atten. coeff., CiteSeer, class 2

(f) Node class., CiteSeer, class 2

(g) Atten. coeff., CiteSeer, class 3

(h) Node class., CiteSeer, class 3

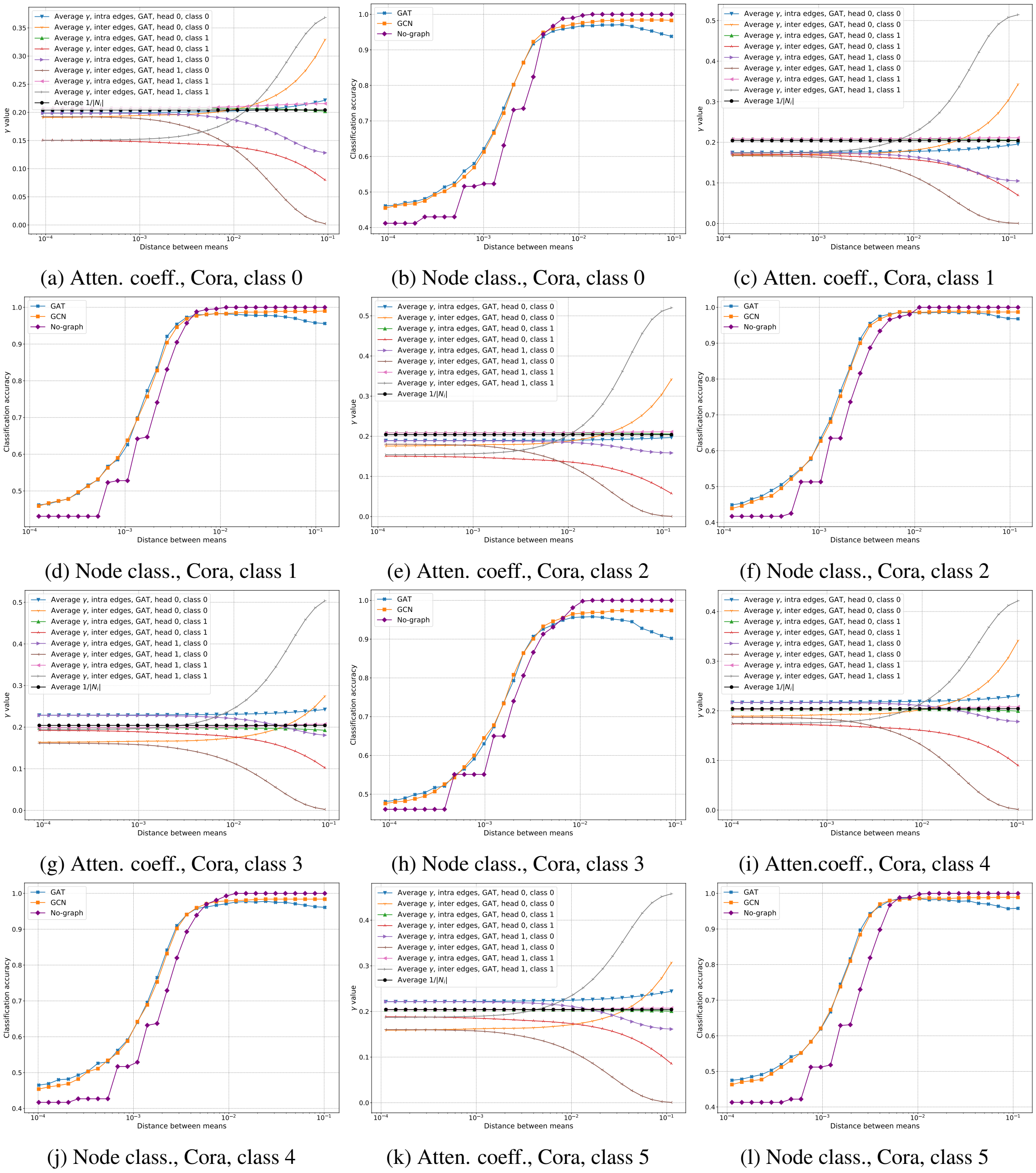
(i) Atten. coeff., CiteSeer, class 4

(j) Node class., CiteSeer, class 4

(k) Atten. coeff., CiteSeer, class 5

(l) Node class., CiteSeer, class 5

Figure 10: Ansatz GAT on CiteSeer.



(a) Atten. coeff., Cora, class 0

(b) Node class., Cora, class 0

(c) Atten. coeff., Cora, class 1

(d) Node class., Cora, class 1

(e) Atten. coeff., Cora, class 2

(f) Node class., Cora, class 2

(g) Atten. coeff., Cora, class 3

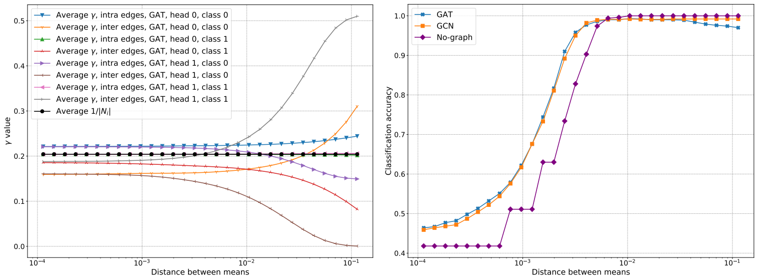
(h) Node class., Cora, class 3

(i) Atten.coeff., Cora, class 4

(j) Node class., Cora, class 4

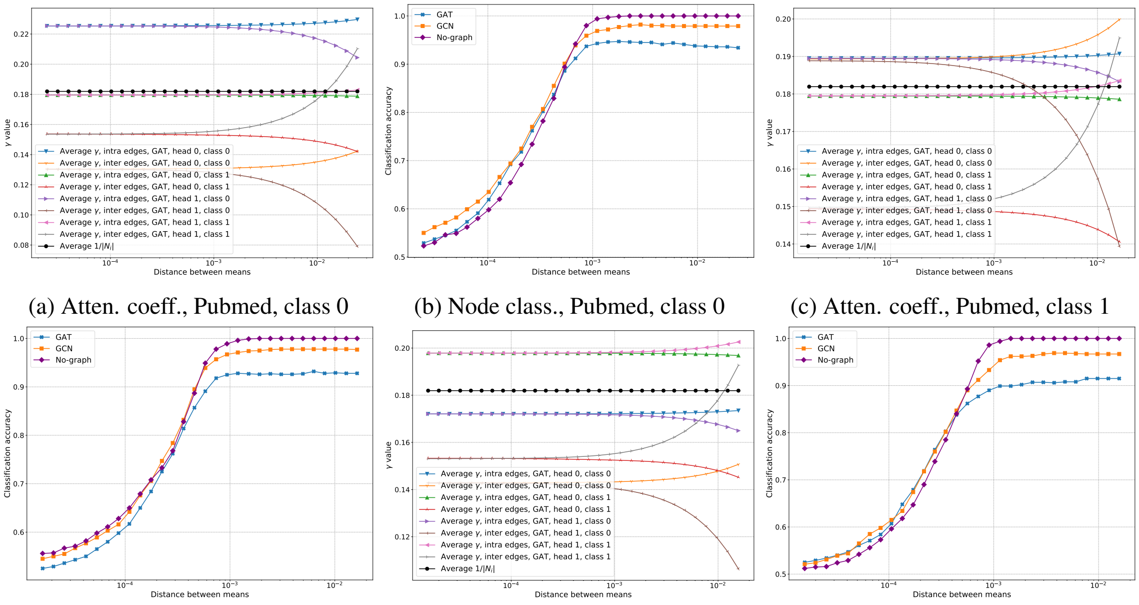
(k) Atten. coeff., Cora, class 5

(l) Node class., Cora, class 5



(m) Atten. coeff., Cora, class 6 (n) Node class., Cora, class 6

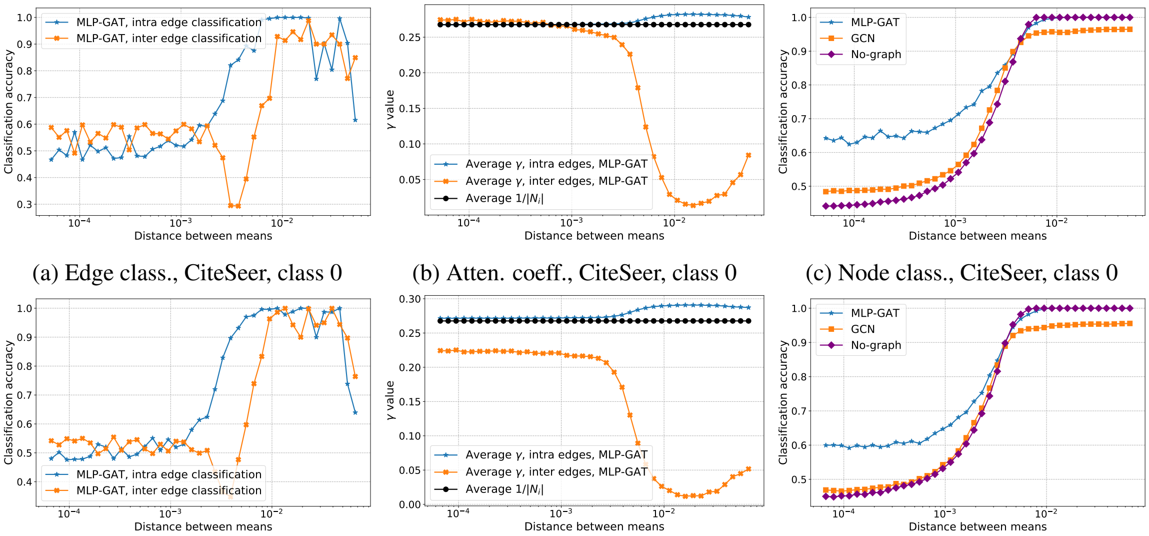
Figure 11: Ansatz GAT on Cora.



(a) Atten. coeff., Pubmed, class 0 (b) Node class., Pubmed, class 0 (c) Atten. coeff., Pubmed, class 1

(d) Node class., Pubmed, class 1 (e) Atten. coeff., Pubmed, class 2 (f) Node class., Pubmed, class 2

Figure 12: Ansatz GAT on PubMed.



(a) Edge class., CiteSeer, class 0

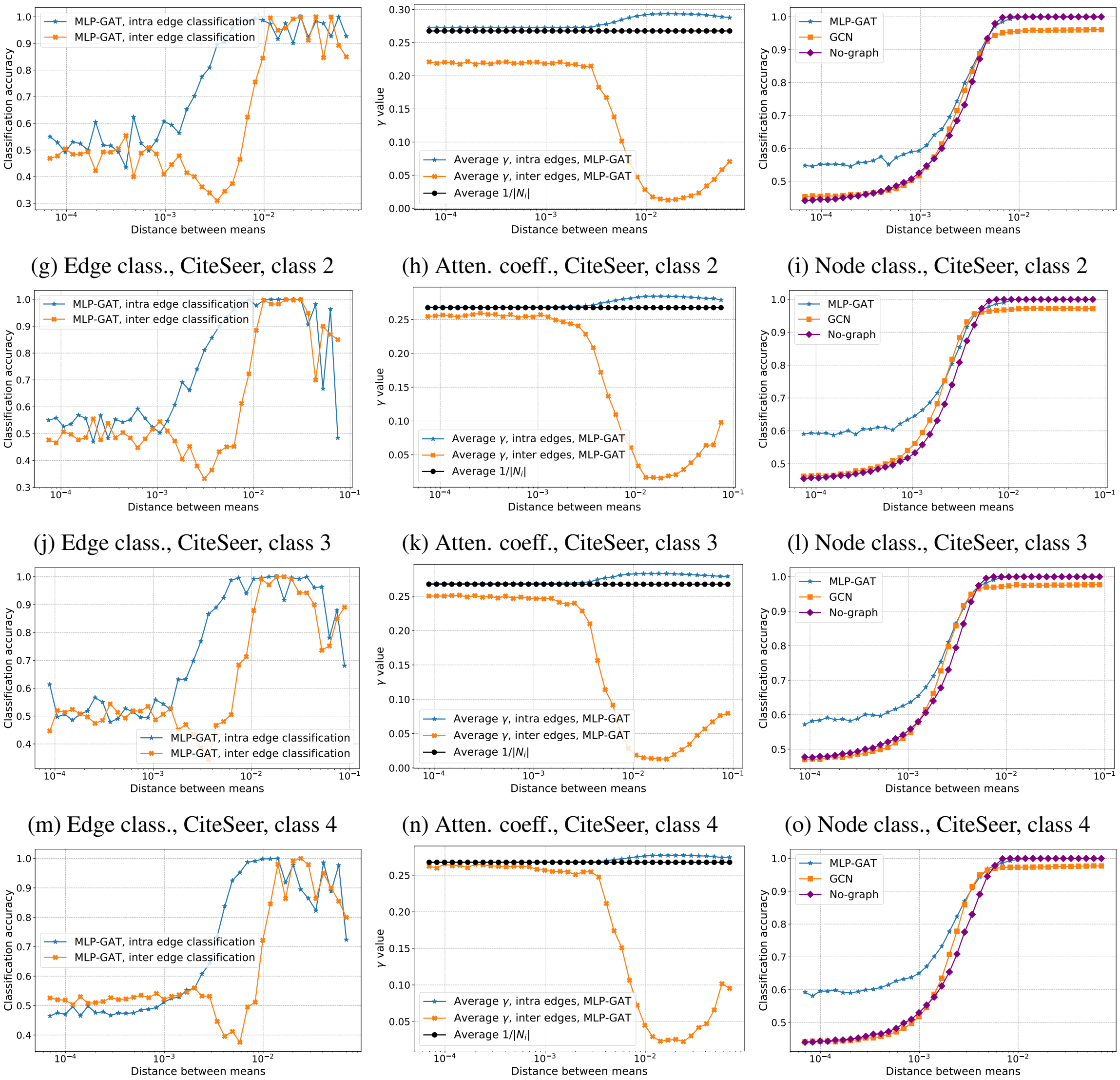
(b) Atten. coeff., CiteSeer, class 0

(c) Node class., CiteSeer, class 0

(d) Edge class., CiteSeer, class 1

(e) Atten. coeff., CiteSeer, class 1

(f) Node class., CiteSeer, class 1



(g) Edge class., CiteSeer, class 2

(h) Atten. coeff., CiteSeer, class 2

(i) Node class., CiteSeer, class 2

(j) Edge class., CiteSeer, class 3

(k) Atten. coeff., CiteSeer, class 3

(l) Node class., CiteSeer, class 3

(m) Edge class., CiteSeer, class 4

(n) Atten. coeff., CiteSeer, class 4

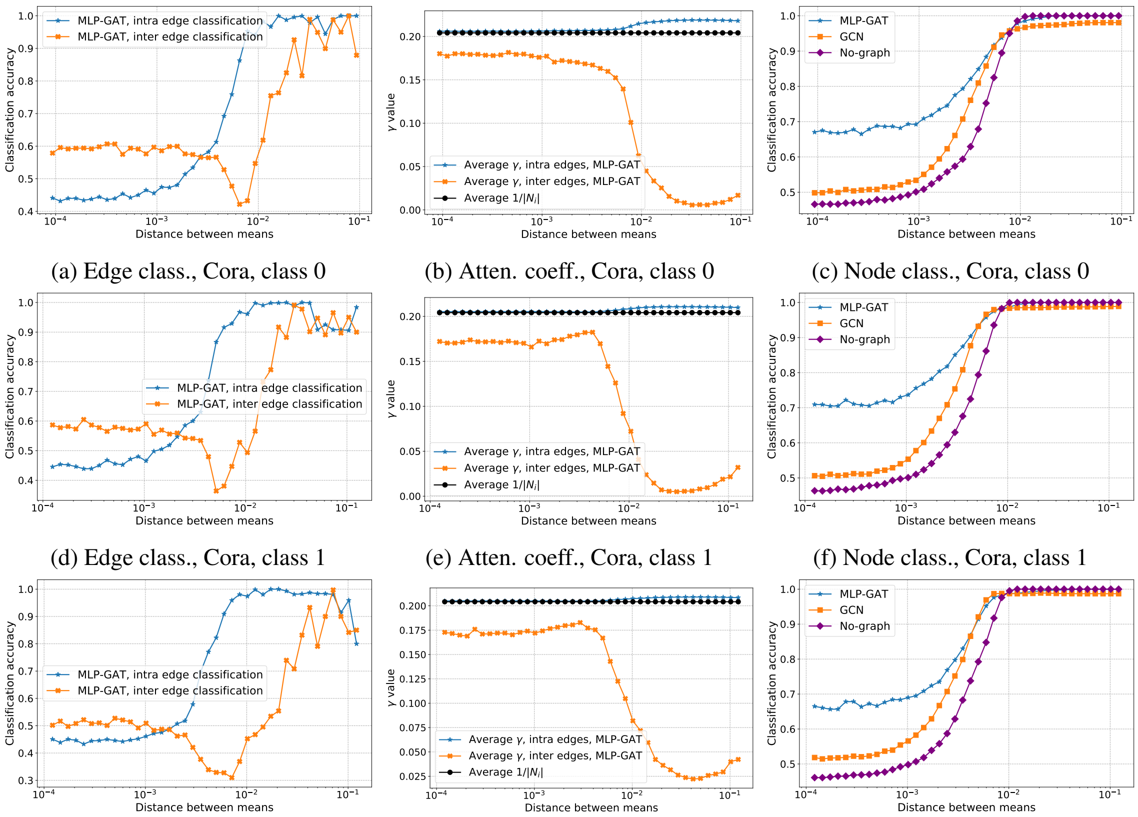
(o) Node class., CiteSeer, class 4

(p) Edge class., CiteSeer, class 5

(q) Atten. coeff., CiteSeer, class 5

(r) Node class., CiteSeer, class 5

Figure 13: Training MLP-GAT on CiteSeer.



(a) Edge class., Cora, class 0

(b) Atten. coeff., Cora, class 0

(c) Node class., Cora, class 0

(d) Edge class., Cora, class 1

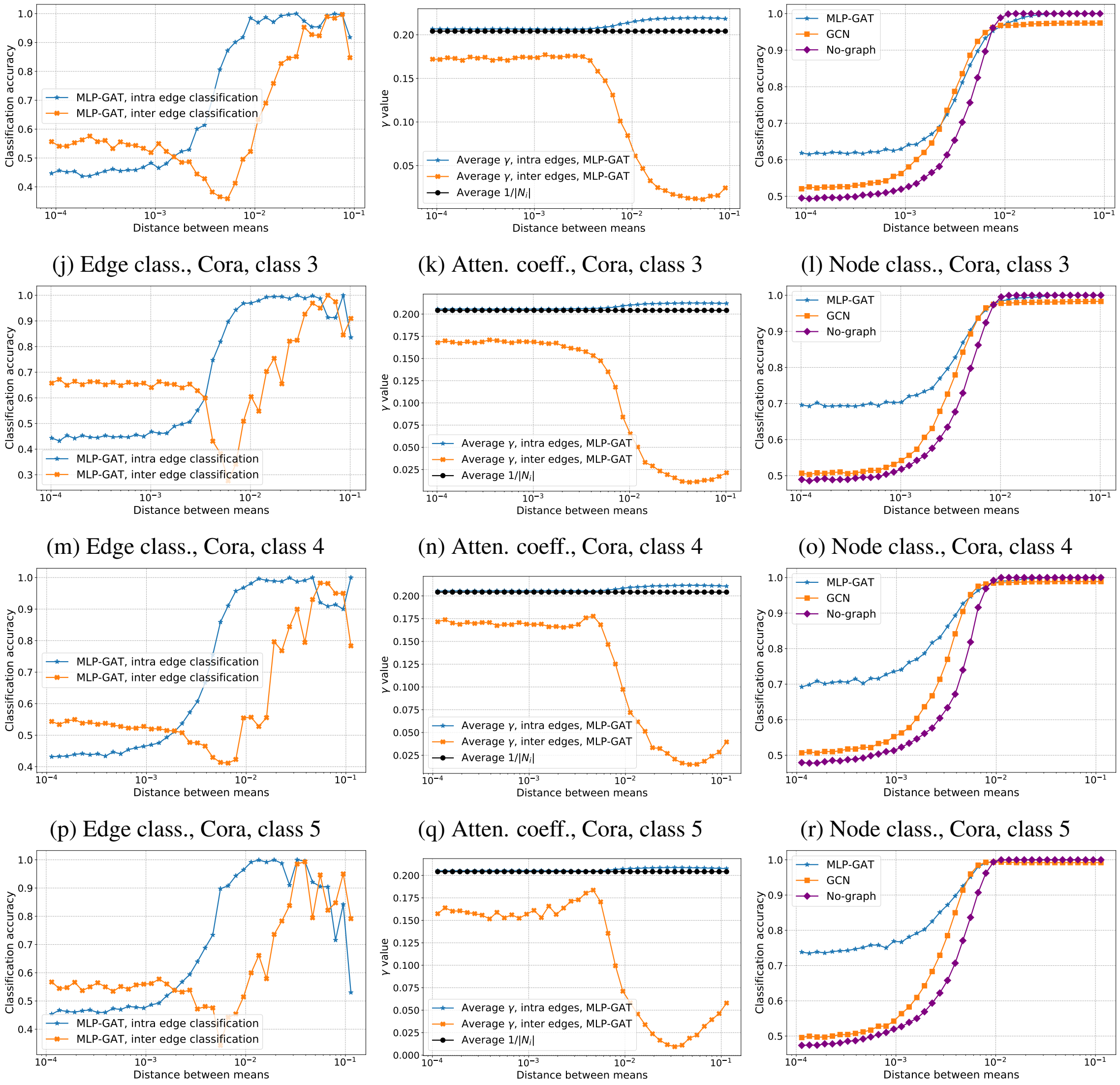
(e) Atten. coeff., Cora, class 1

(f) Node class., Cora, class 1

(g) Edge class., Cora, class 2

(h) Atten. coeff., Cora, class 2

(i) Node class., Cora, class 2



(j) Edge class., Cora, class 3

(k) Atten. coeff., Cora, class 3

(l) Node class., Cora, class 3

(m) Edge class., Cora, class 4

(n) Atten. coeff., Cora, class 4

(o) Node class., Cora, class 4

(p) Edge class., Cora, class 5

(q) Atten. coeff., Cora, class 5

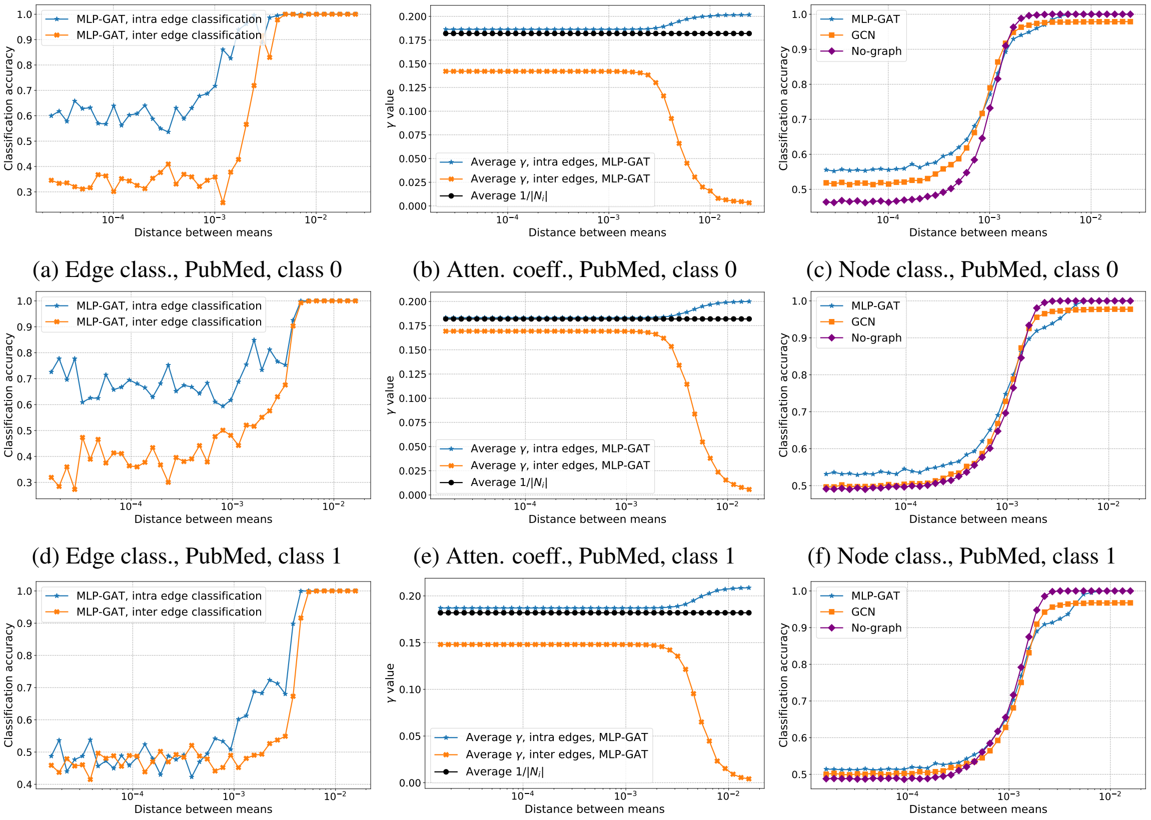
(r) Node class., Cora, class 5

(s) Edge class., Cora, class 6

(t) Atten. coeff., Cora, class 6

(u) Node class., Cora, class 6

Figure 14: Training MLP-GAT on Cora.



(a) Edge class., PubMed, class 0

(b) Atten. coeff., PubMed, class 0

(c) Node class., PubMed, class 0

(d) Edge class., PubMed, class 1

(e) Atten. coeff., PubMed, class 1

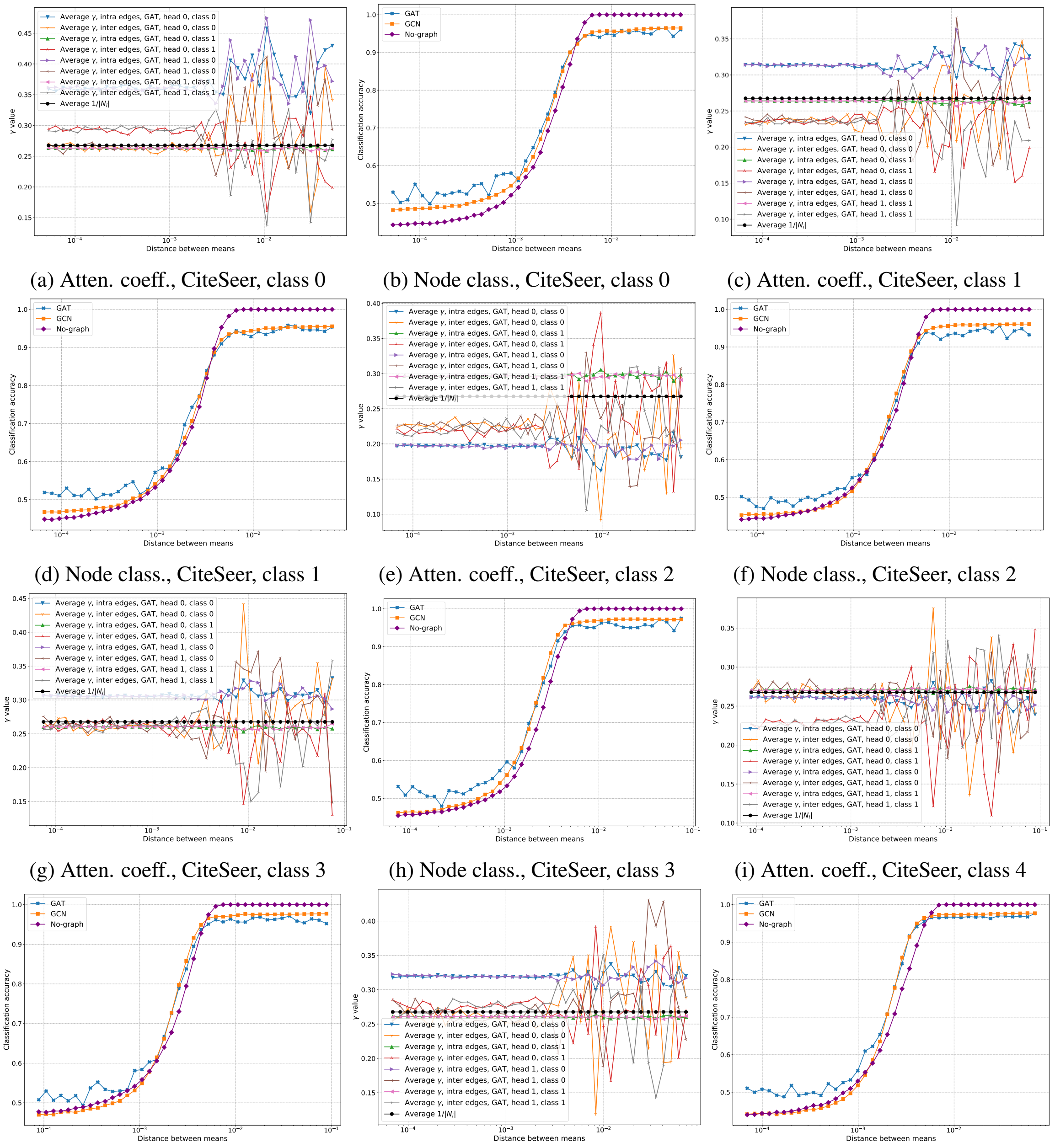
(f) Node class., PubMed, class 1

(g) Edge class., PubMed, class 2

(h) Atten. coeff., PubMed, class 2

(i) Node class., PubMed, class 2

Figure 15: Training MLP-GAT on PubMed.



(a) Atten. coeff., CiteSeer, class 0

(b) Node class., CiteSeer, class 0

(c) Atten. coeff., CiteSeer, class 1

(d) Node class., CiteSeer, class 1

(e) Atten. coeff., CiteSeer, class 2

(f) Node class., CiteSeer, class 2

(g) Atten. coeff., CiteSeer, class 3

(h) Node class., CiteSeer, class 3

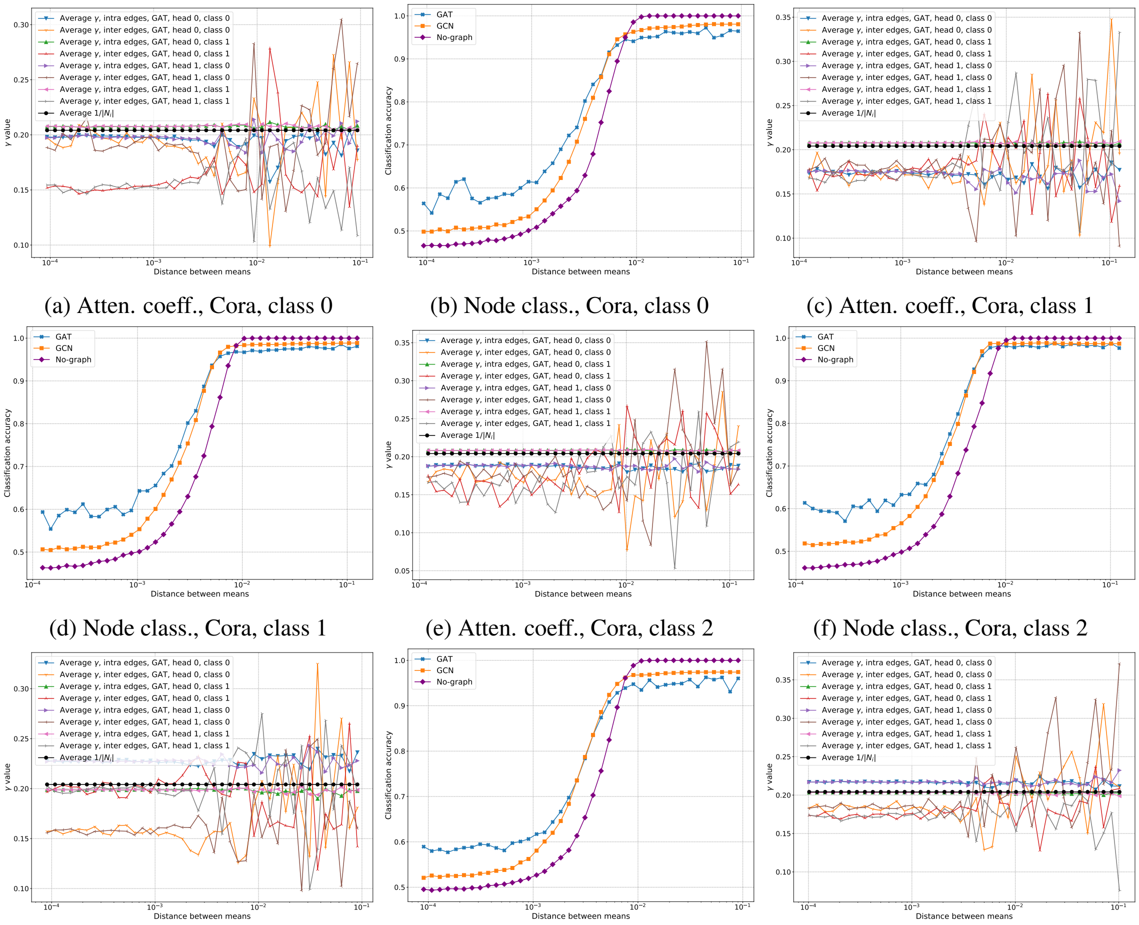
(i) Atten. coeff., CiteSeer, class 4

(j) Node class., CiteSeer, class 4

(k) Atten. coeff., CiteSeer, class 5

(l) Node class., CiteSeer, class 5

Figure 16: Training GAT on CiteSeer.



(a) Atten. coeff., Cora, class 0

(b) Node class., Cora, class 0

(c) Atten. coeff., Cora, class 1

(d) Node class., Cora, class 1

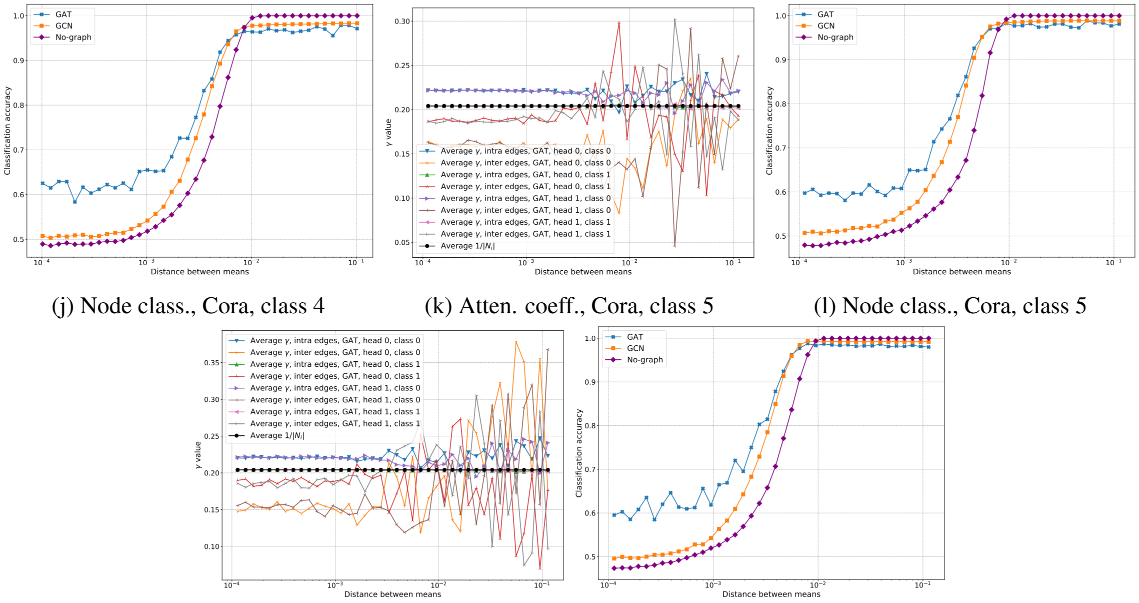
(e) Atten. coeff., Cora, class 2

(f) Node class., Cora, class 2

(g) Atten. coeff., Cora, class 3

(h) Node class., Cora, class 3

(i) Atten. coeff., Cora, class 4



(j) Node class., Cora, class 4

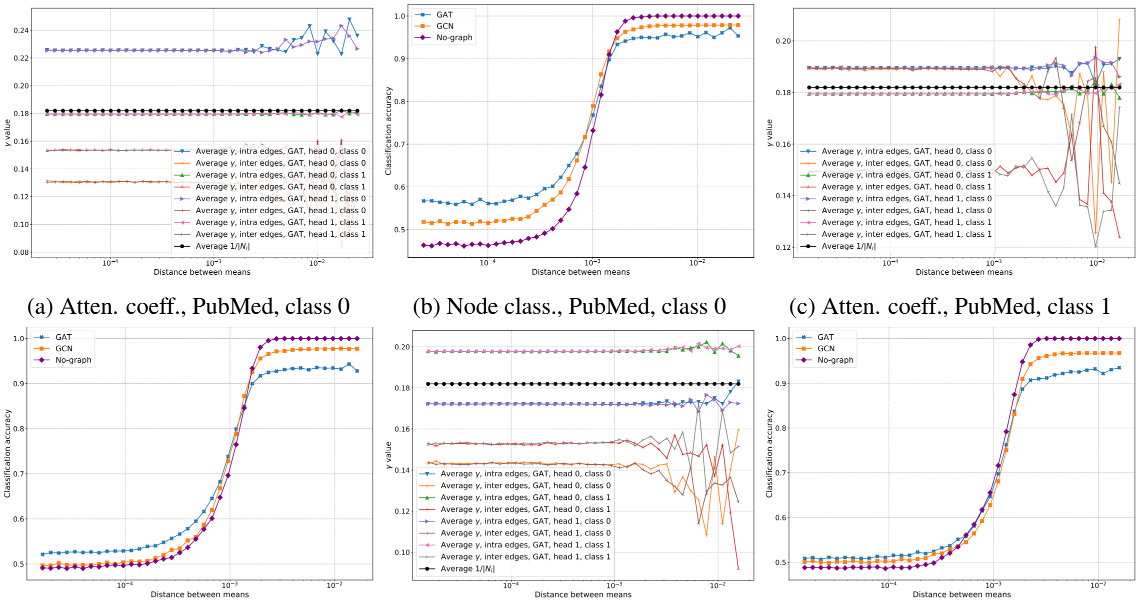
(k) Atten. coeff., Cora, class 5

(l) Node class., Cora, class 5

(m) Atten. coeff., Cora, class 6

(n) Node class., Cora, class 6

Figure 17: Training GAT on Cora.



(a) Atten. coeff., PubMed, class 0

(b) Node class., PubMed, class 0

(c) Atten. coeff., PubMed, class 1

(d) Node class., PubMed, class 1

(e) Atten. coeff., PubMed, class 2

(f) Node class., PubMed, class 2

Figure 18: Training GAT on PubMed.

C FREQUENTLY ASKED QUESTIONS AND ANSWERS

In this section we provide answers to some questions typically asked by practitioners working with graph neural networks.

Q: *The regimes are defined based on the relationship between population parameters $\boldsymbol{\mu}$ and σ. Why don't you define the regimes based on empirical/sample mean and variance? Moreover, why does the definition of regimes depend on the sample size n?*

A: For the first question: Our goal is to understand how graph attention performs with respect to the true data distribution parameters μ and σ. These parameters naturally appear in the analyses because we can replace the random variables $\mathbf{X}$ with $\pm\boldsymbol{\mu} + \sigma\boldsymbol{g}$ where $\boldsymbol{g}$ is isotropic Gaussian. This replacement is exact. Orthogonal to our setting, one may try and introduce in the analyses the empirical mean and variance, but that would result in additional errors in the bounds which are not necessary, since we can already prove high probability results using the true parameters. Therefore the correct way to make assumptions is to assume something about the population parameters. Results under our assumptions provide a cleaner theoretical insight because the same conclusions apply to any realized empirical data with high probability. In practice if one is interested in determining if the assumptions are satisfied in the real data, then it could be posed as a parameter estimation problem where the estimation errors for both mean and variance can be controlled.
For the second question: The definition of regimes depends on n because the results require Gaussian concentration. In fact, they should depend on n. Here is an intuitive explanation. Think about what happens if $n = 2$, then any model will work. The larger n is, the larger ratio of mean over variance we require in order to avoid overlap between the Gaussian distributions. Note that the dependence on n is unavoidable if we want to obtain high probability result on correct classification: The assumptions and the result are essentially an "if and only if" relationship.

Q: *In the "easy regime", are the solutions given by SGD or other optimizers (close to) optimal?*

A: Empirically, we have extensive experiments (see Figures 13-18) where we demonstrate that training the models using SGD is doing what it is expected to do, i.e., the trained models and the ansatz have similar performances. Note that the ansatz is proven to be optimal for the "easy regime".

Q: *Are the theoretical analyses restricted to a specific attention architecture?*

A: No. Our analyses are not limited to a special graph attention model. For the positive result in the "easy regime", we use a specific architecture because it is sufficient for perfect classification. For the negative result in the "hard regime", our result applies to the Bayes optimal classifier, which means that any classifier/attention would fail (since the optimal Bayes classifier fails).

Q: *Are the theoretical analyses of CSBM in this work similar to prior analyses of SBM?*

A: No. The analyses in this work significantly differ from prior analyses of SBM in that our work heavily focuses on analyzing the behaviors of sub-Gaussian random variables (i.e. the node features) rather than analyzing the graph structure alone. The sub-Gaussian random variables are formed by coupling the highly nonlinear attention coefficients, the node features and the graph structure altogether. The resulting data is highly correlated and highly nonlinear.

Q: *What is the main challenge in extending the current work to multi-layer scenario?*

A: For the particular two-block CSBM data model, we do not need more layers to obtain the desired results for both the "easy regime" and the "hard regime". On the other hand, if someone wants to analyze the effects of using more layers, they need to use a more complicated data model, e.g. XOR CSBM, where the nodes belonging to class 0 have means $[\boldsymbol{\mu}, \boldsymbol{\mu}]$ and $[-\boldsymbol{\mu}, -\boldsymbol{\mu}]$ with equal probability, and nodes belonging to class 1 have means $[\boldsymbol{\mu}, -\boldsymbol{\mu}]$ and $[-\boldsymbol{\mu}, \boldsymbol{\mu}]$ with equal probability. For XOR CSBM we might need more than two layers. The proof techniques are going to be the same, and conclusions are going to be the same.

Q: *Ultimately we care about node classification accuracy, why do you focus on edge classification in this work?*

A: The ability to weight/drop edges allows aggregation from more informative neighbor nodes and consequently leads to better node classification result. This benefit is demonstrated by the positive result in the "easy" regime. For that reason we believe it is important to understand the behavior of graph attention from the perspective of edge classification. This is why in this work we focus on determining if graph attention helps in the classification of edges.