Demystifying Structural Disparity in Graph Neural Networks: Can one size Fit ALL?

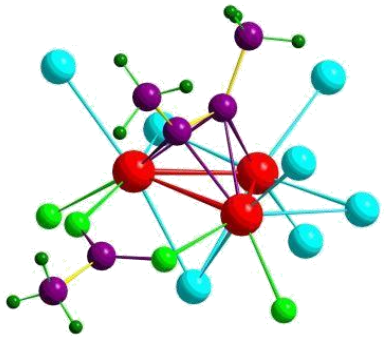
Haitao Mao, Zhikai Chen, Wei Jin, Haoyu Han
Yao Ma, Tong Zhao, Neil Shah, Jiliang Tang

Department of Computer Science and Engineering
Michigan State University
haitaoma@msu.edu

June, 2023

Graph data are everywhere

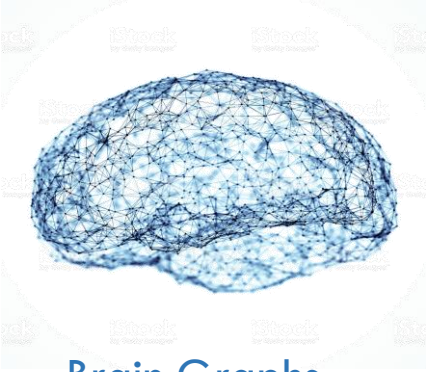
Social Graphs



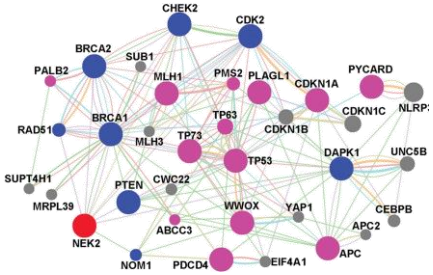
Molecular Graphs

Transportation Graphs

Web Graphs



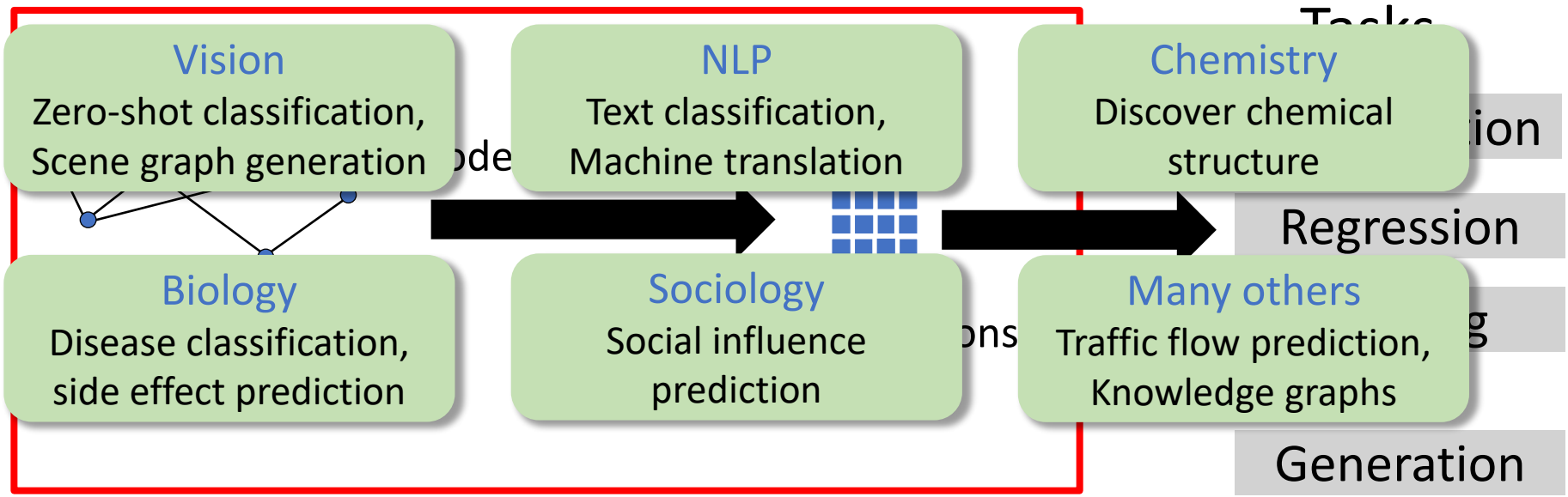
Brain Graphs



Gene Graphs

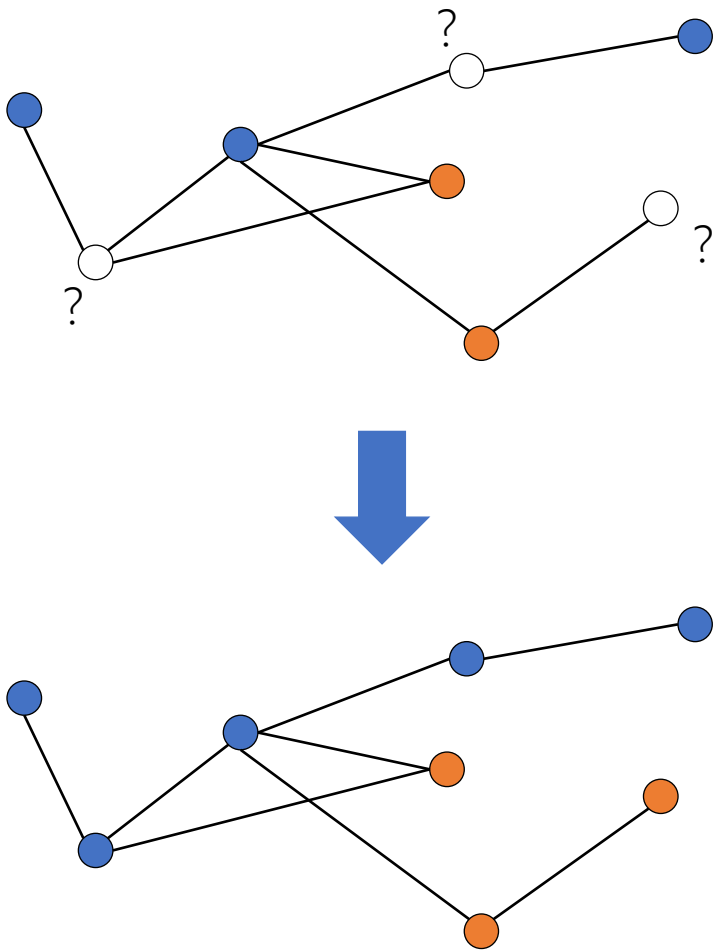
Representation learning on Graphs

New frontier beyond classic ML that only learns on images and sequences



Representation Learning on Graphs

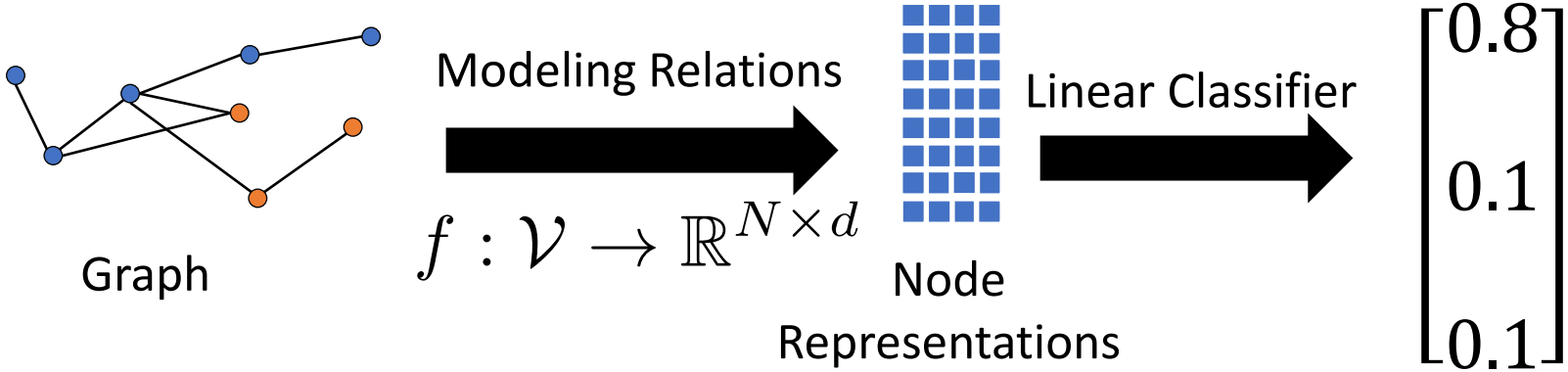
Node Classification task



- Inferring Node Attributes
- Social Influence Prediction
- Traffic Prediction
- Air Quality Prediction

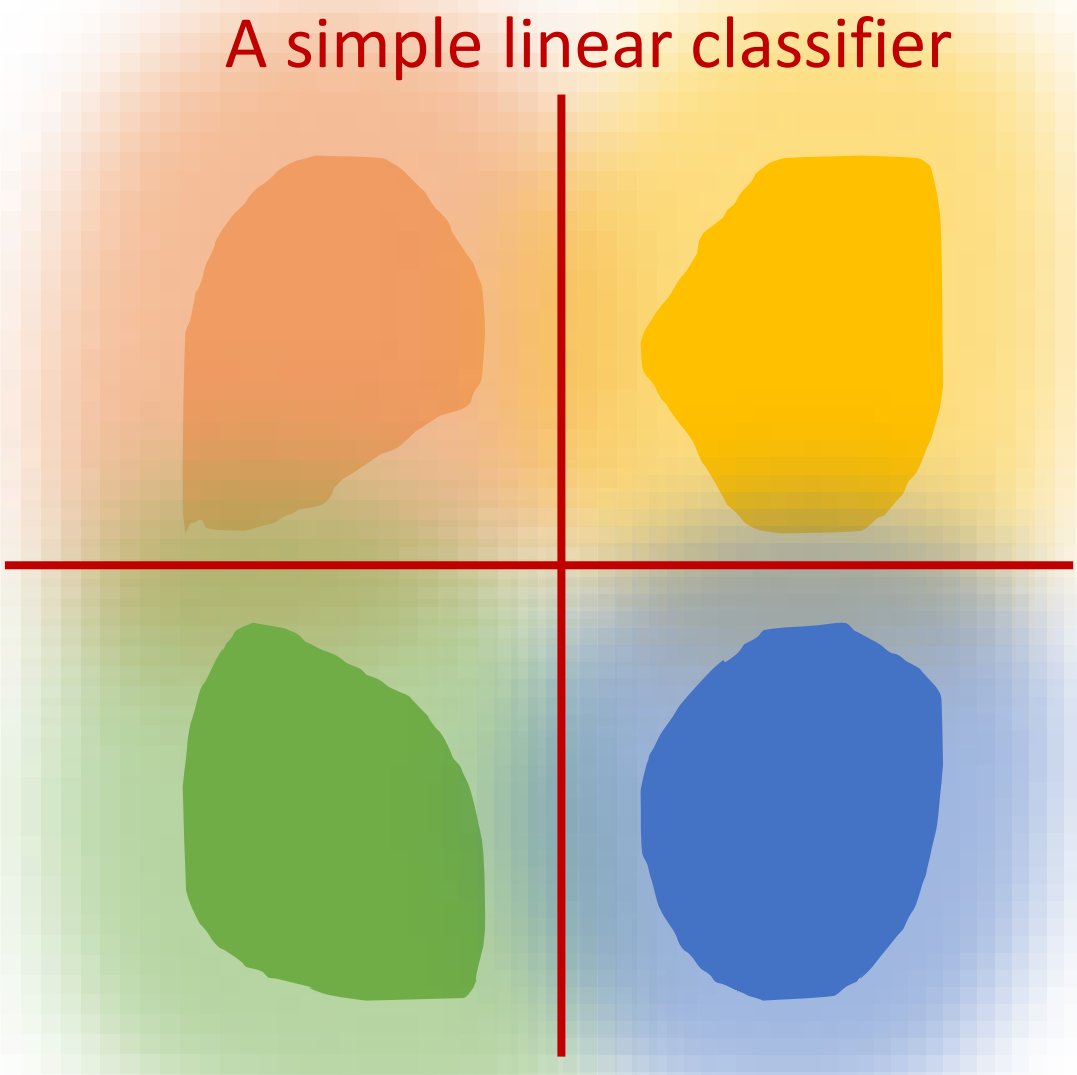
⋮

Representation learning for Node Classification?



What is a good node representation?

What is a good node representation?



Cohesion:
Intra-class similar

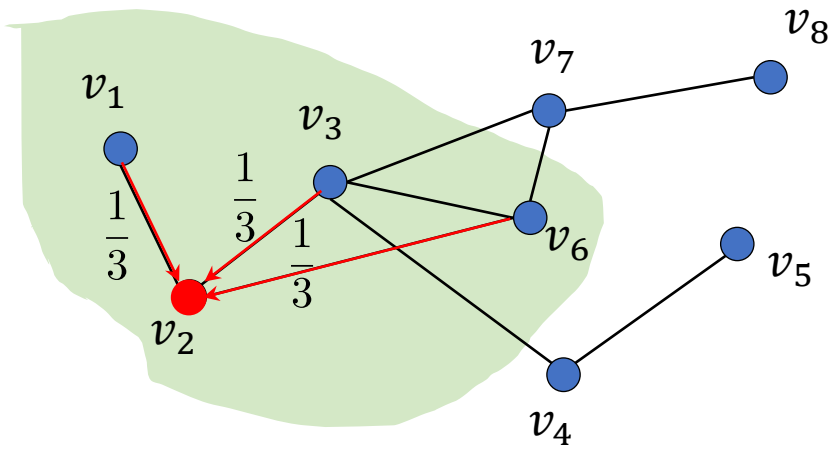
Separation:
Inter-class dissimilar

Graph Neural Networks

Neighbors of node v_2

$$\mathcal{N}(2)$$

$$\{v_1, v_3, v_6\}$$

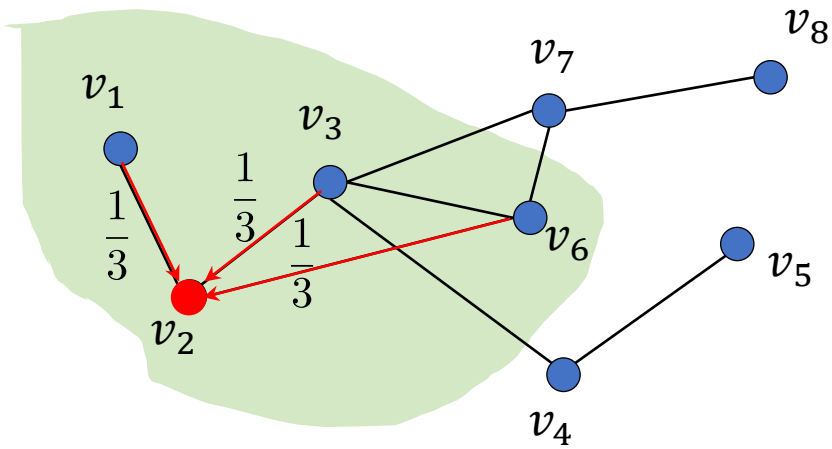


Graph Neural Networks

Neighbors of node v_2

$\mathcal{N}(2)$

$\{v_1, v_3, v_6\}$



Feature Transformation

GNN:

$$\mathbf{F}_i = \boxed{\sum_{j \in \mathcal{N}(i)} \frac{1}{|\mathcal{N}(i)|}} \boxed{\mathbf{X}_j \mathbf{W}}$$

$$\mathbf{X}'_j = \mathbf{X}_j \mathbf{W}$$

Neighborhood Aggregation

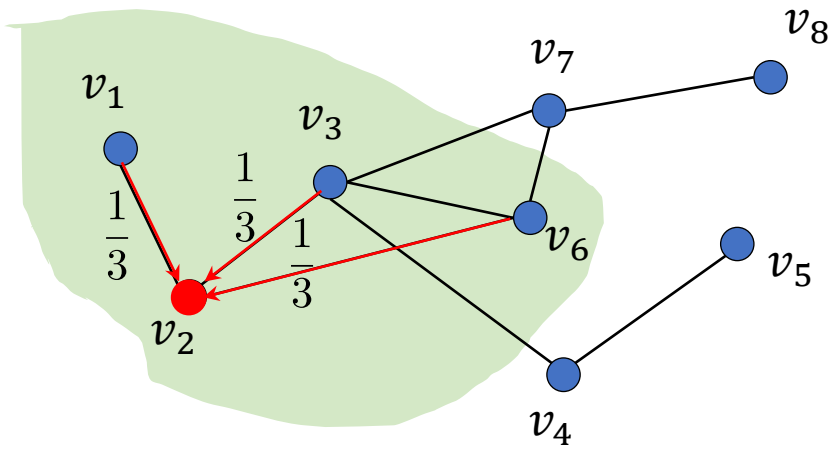
8

Graph Neural Networks

Neighbors of node v_2

$\mathcal{N}(2)$

$\{v_1, v_3, v_6\}$



GNN:

$$\mathbf{F}_i = \boxed{\sum_{j \in \mathcal{N}(i)} \frac{1}{|\mathcal{N}(i)|}} \mathbf{X}'_j$$

Majorly focusing on Aggregation

Neighborhood Aggregation

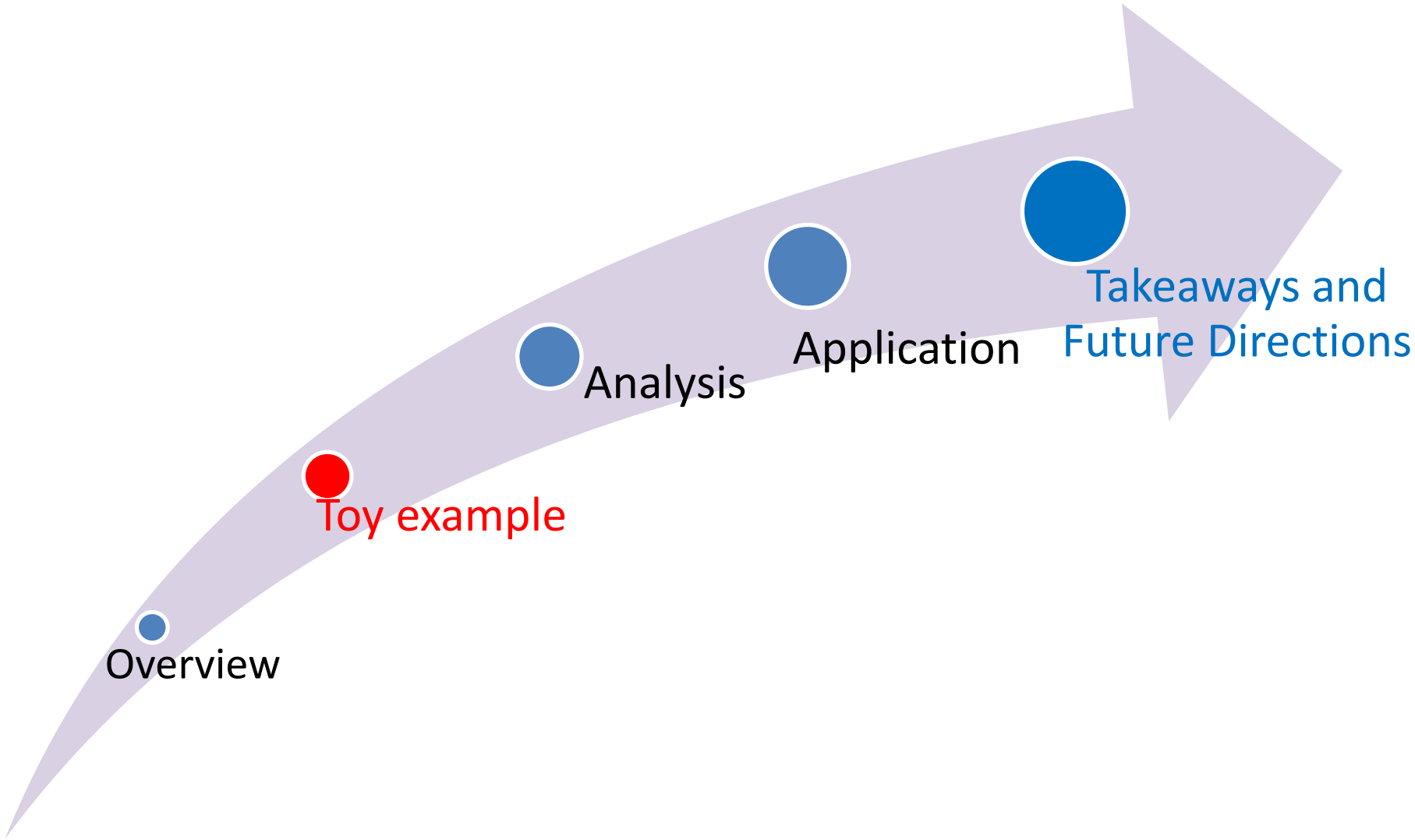
This talk

Can GNNs learn a good representation for all the nodes?

When can GNNs show good node classification performance?

No, GNNs may even underperform MLP

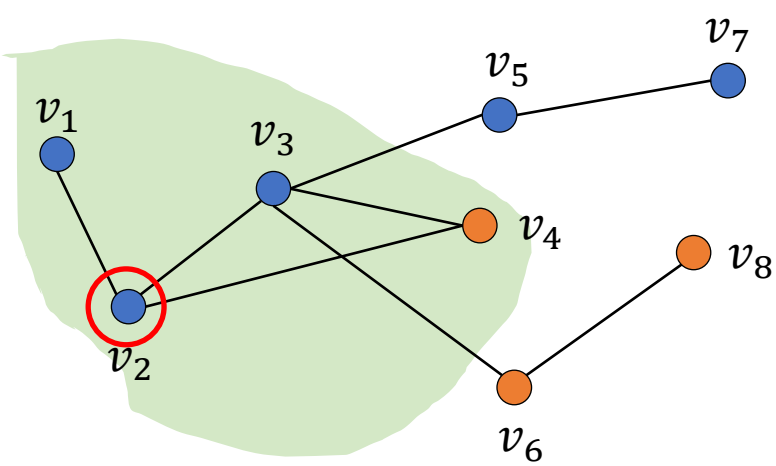
Outline


Overview
Toy example
Analysis
Application
Takeaways and
Future Directions

Preliminary: homophily & heterophily

Homophily: "nodes tend to connect with "similar" or "alike" others"

$$\mathrm{h}_i = \frac{|\{u \in \mathcal{N}(v_i): y_u = y_v\}|}{|\mathcal{N}(v_i)|}$$



Center node: v_2

Neighbor set: v_1 v_3 v_4

Node homophily ratio of v_2:

$$h_2 = \frac{2}{3}$$

Data assumption for homophily & heterophily

CSBM(μ_0, μ_1, p, q)

Feature generation

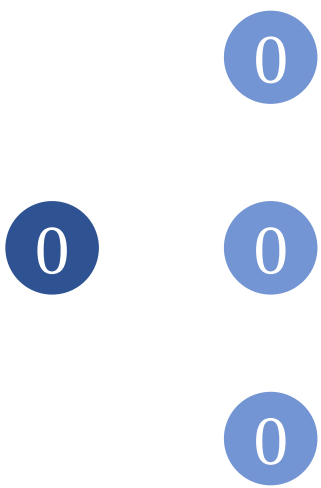
$x_i \sim \mathcal{N}(\mu_0, I)$ For c_j = 0

0

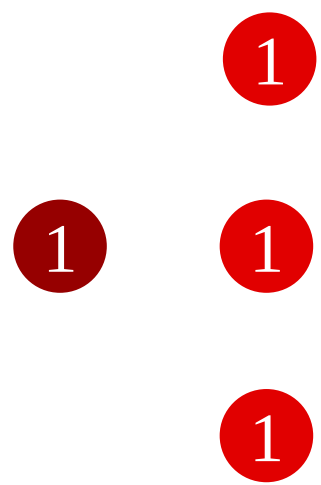
$x_j \sim \mathcal{N}(\mu_1, I)$ For c_j = 1

1

Nodes in class 0



Nodes in class 1



Data assumption for homophily & heterophily

CSBM(μ_0, μ_1, p, q)

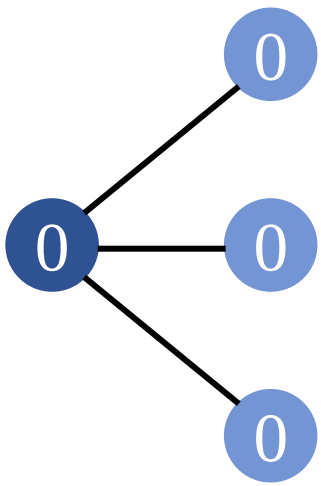
Edge generation

$$\begin{cases} e_{ij} \sim B(1,p) & \text{when } c_i = c_j \\ e_{ij} \sim B(1,q) & \text{when } c_i \neq c_j \end{cases}$$

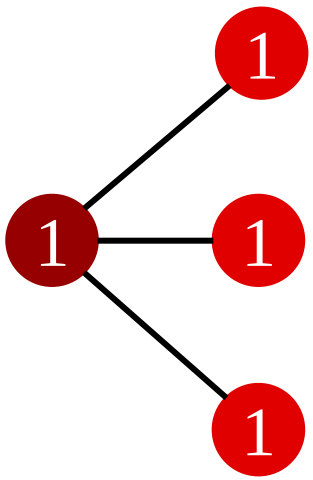
Intra-class probability: $p = 0.8$

Inter-class probability: $q = 0$

Nodes in class 0



Nodes in class 1



A homophily case

Data assumption for homophily & heterophily

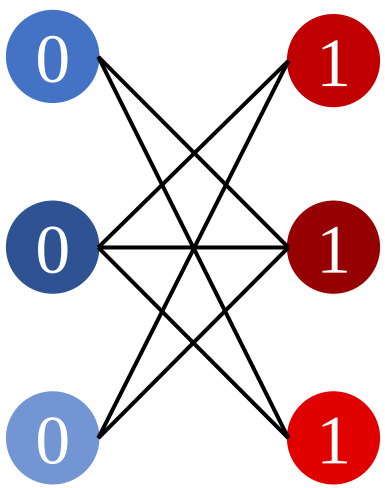
CSBM(μ_0, μ_1, p, q)

Edge generation

$$\begin{cases} e_{ij} \sim B(1, p) & \text{when } c_i = c_j \\ e_{ij} \sim B(1, q) & \text{when } c_i \neq c_j \end{cases}$$

Intra-class probability: $p = 0$

Inter-class probability: $q = 0.8$

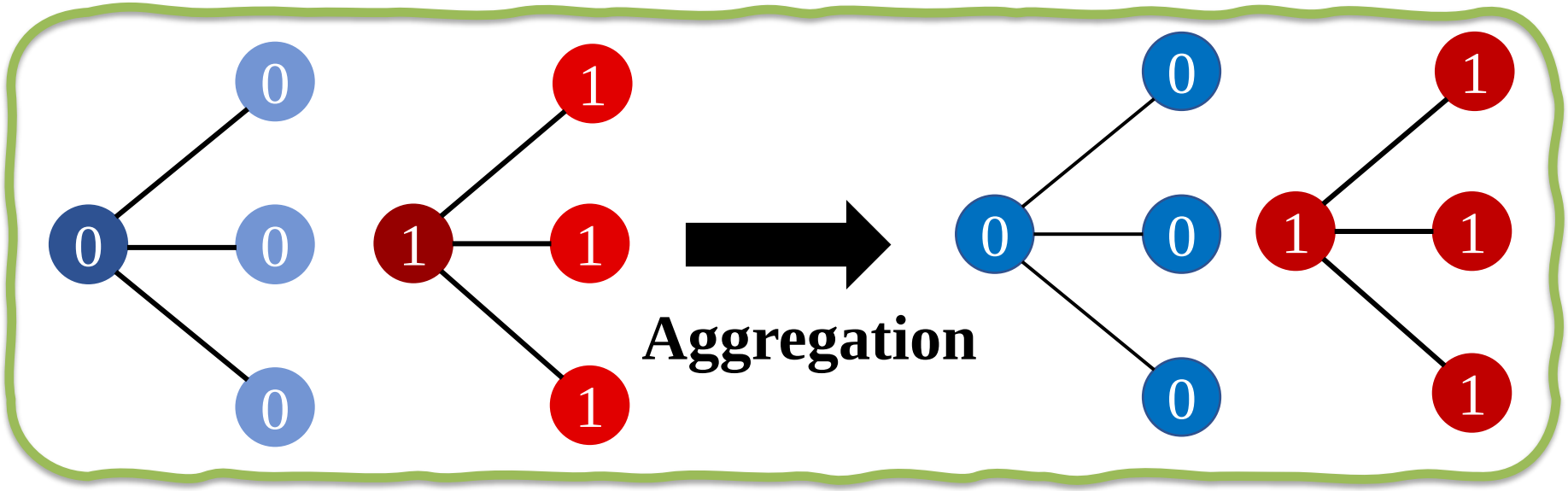


A heterophily case

How can GNN work well on homophily?

The Homophily Case

$$\mathbf{F}_i = \sum_{j \in \mathcal{N}(i)} \frac{1}{|\mathcal{N}(i)|} \mathbf{X}_j$$

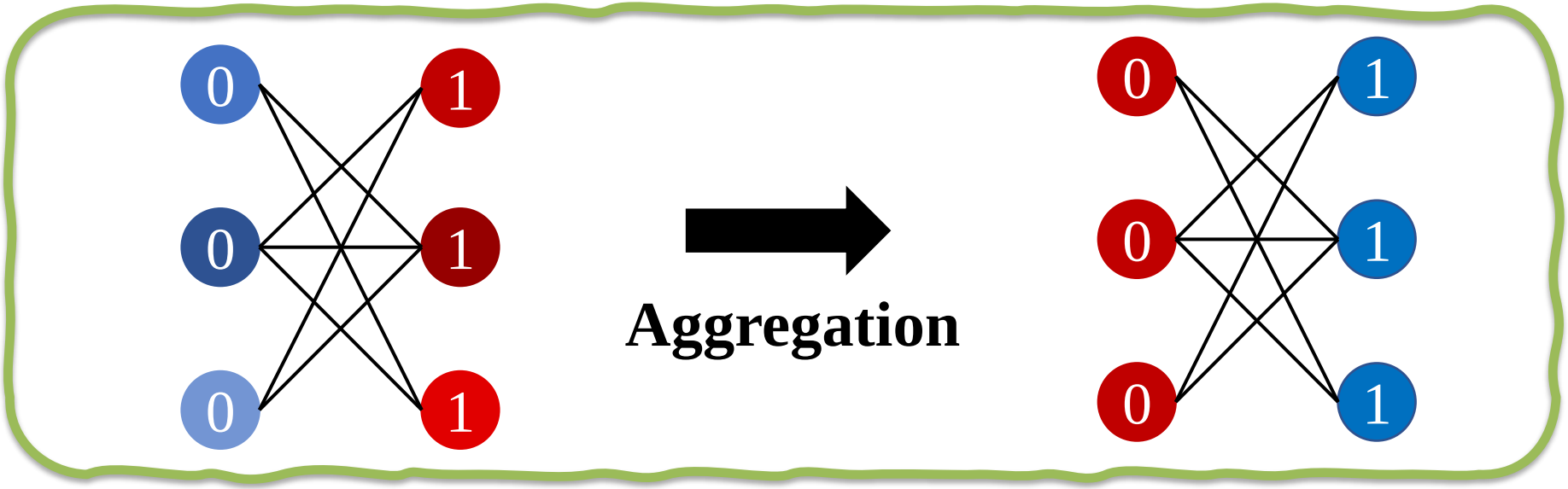


Good feature separability can be observed after aggregation

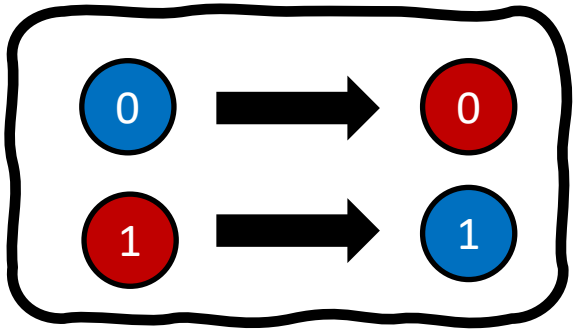
How can GNN work well on heterophily?

The Heterophily Case

$$\mathbf{F}_i = \sum_{j \in \mathcal{N}(i)} \frac{1}{|\mathcal{N}(i)|} \mathbf{X}_j$$



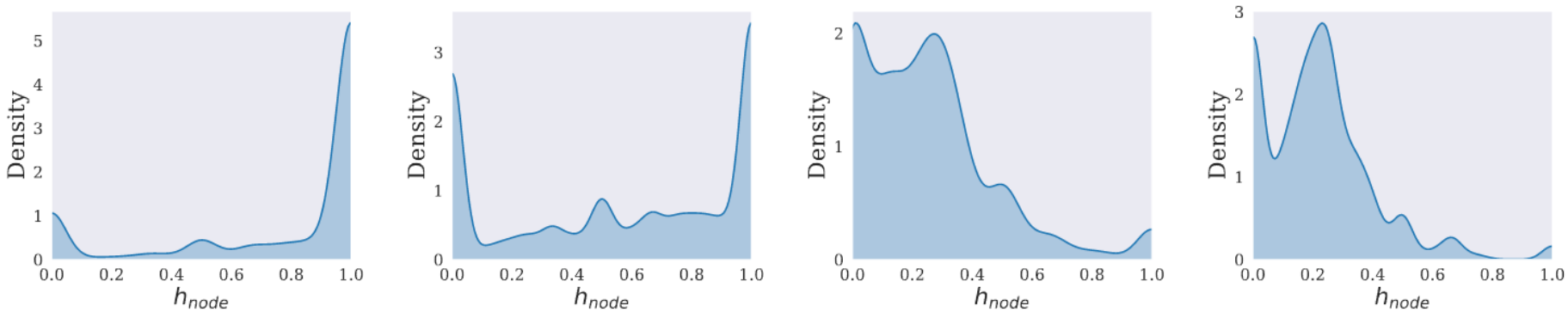
Good separability can still be observed despite alteration



Perfect Separation

Misalign with real-world scenario

Node homophily ratio distribution



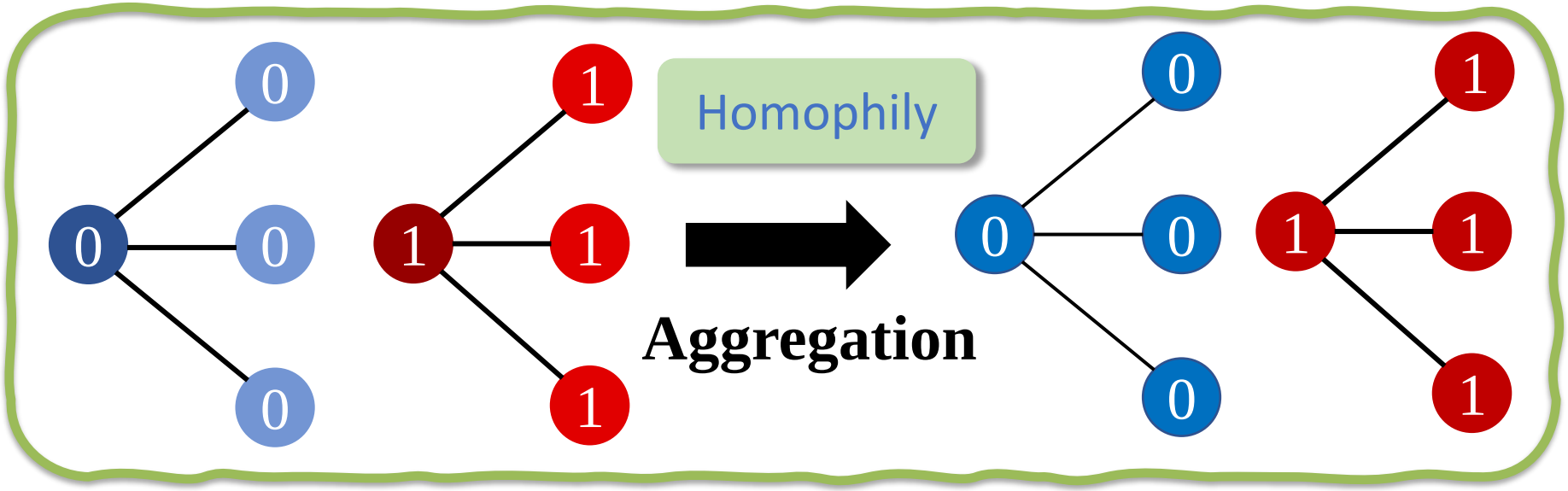
(a) PubMed (h=0.79) (b) Ogbn-arxiv (h=0.63) (c) Chameleon (h=0.22) (d) Squirrel (h=0.25)

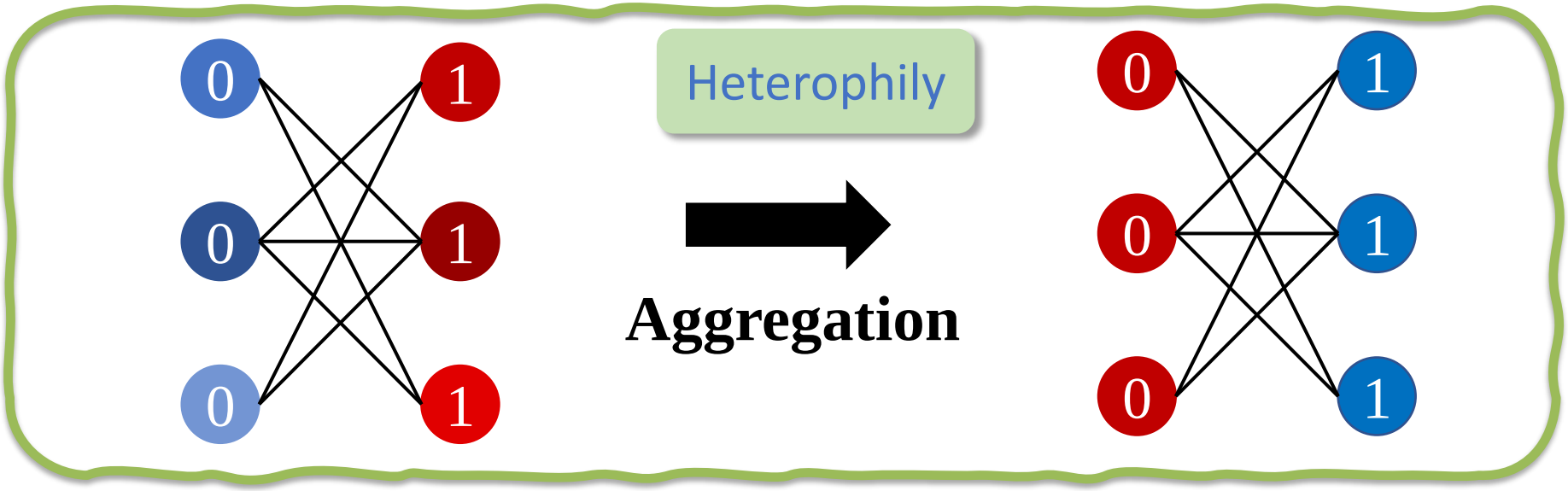
Both homophily and heterophily nodes appears across all real-world graphs

We can not consider homophily or heterophily solely!

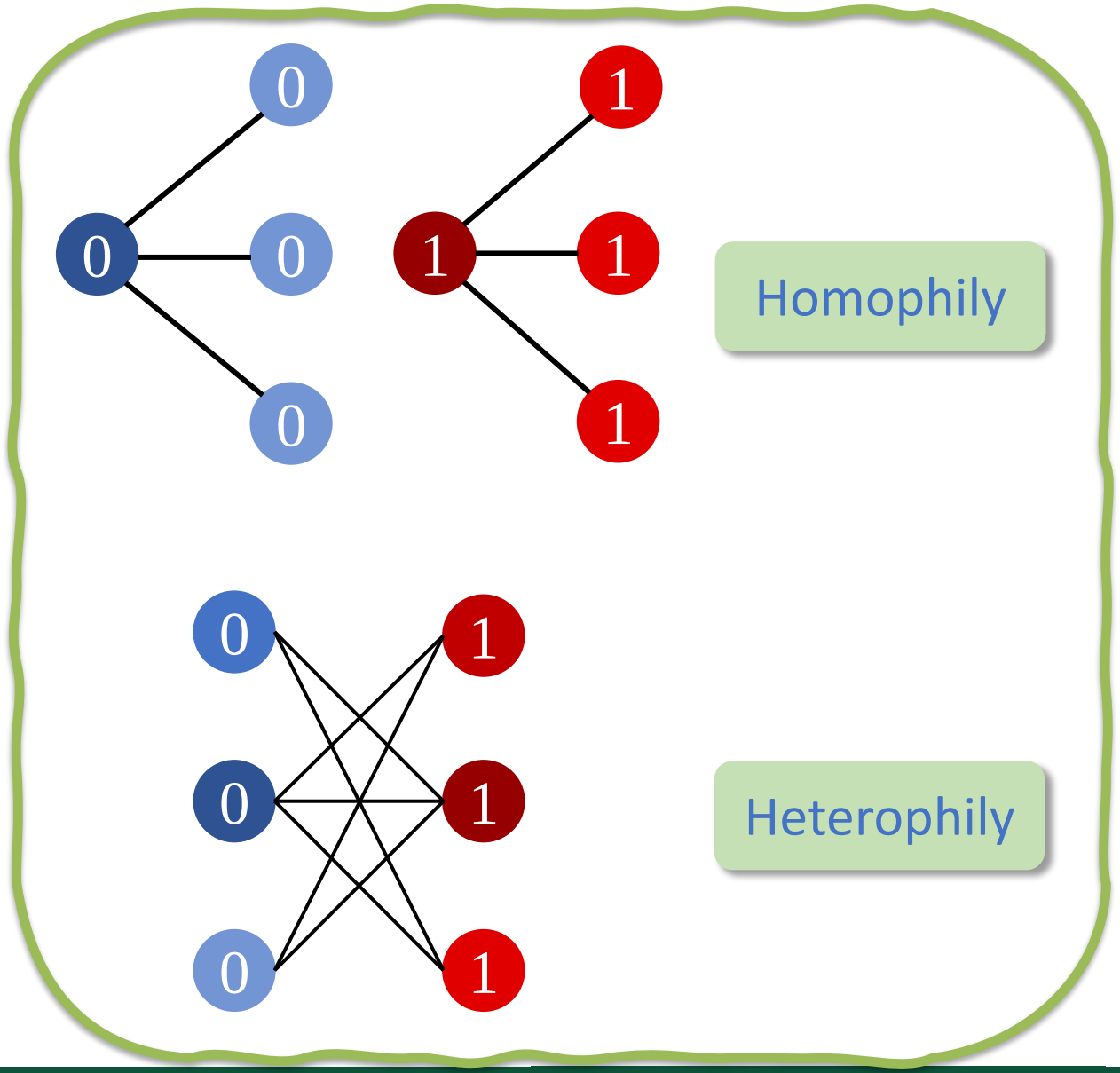
Structure disparity often happens in real world scenario!

How can GNN work well on homophily & heterophily?





How can GNN work well on homophily & heterophily?



Before aggregation

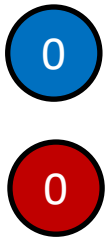
Nodes with class 0 in Blue 0

Nodes with class 1 in Red 1

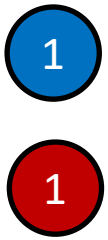
How can GNN work well on homophily & heterophily?

After aggregation

Nodes with class 0 in both Blue and red

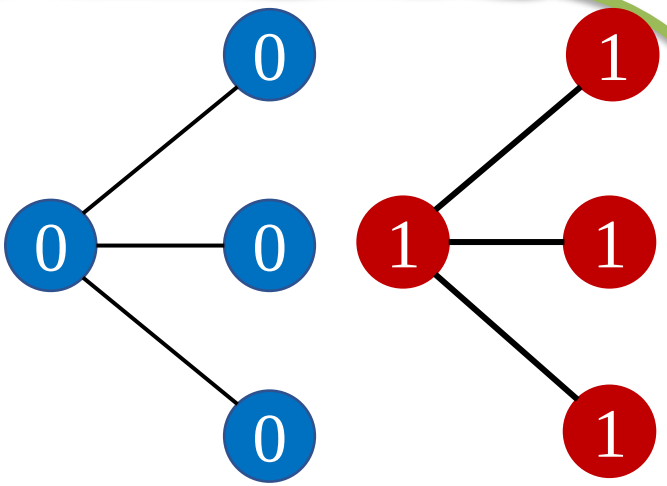


Nodes with class 1 in both Blue and red

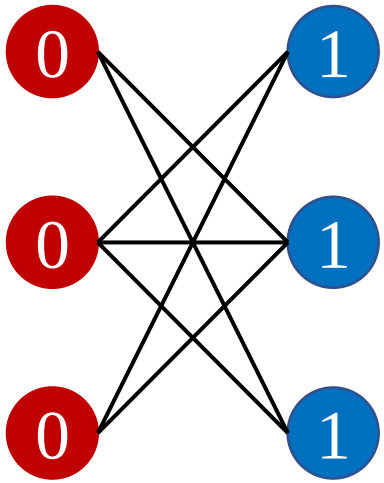


Hard to classify!

Homophily



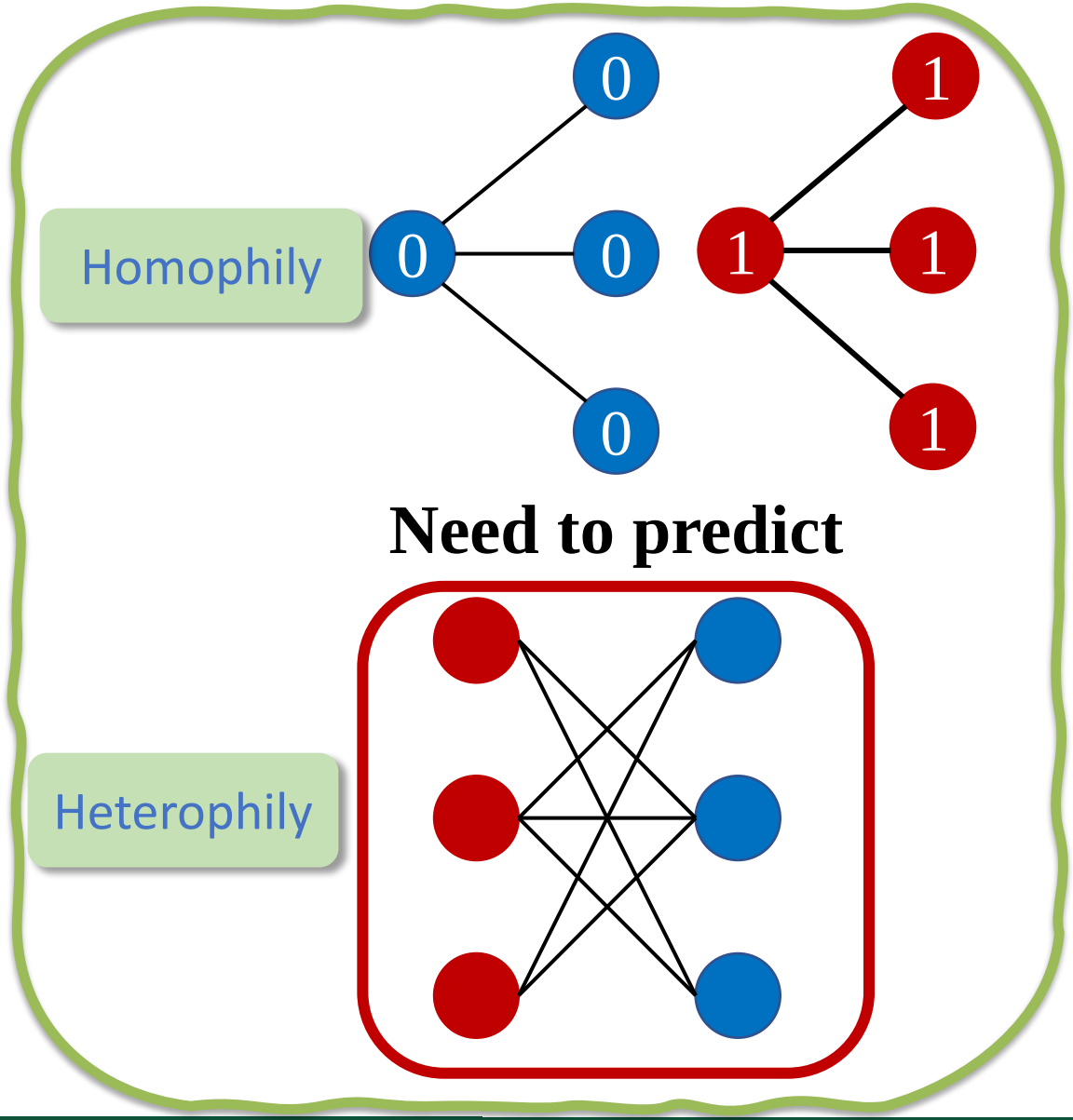
Heterophily



How can GNN work well on homophily & heterophily?

After aggregation

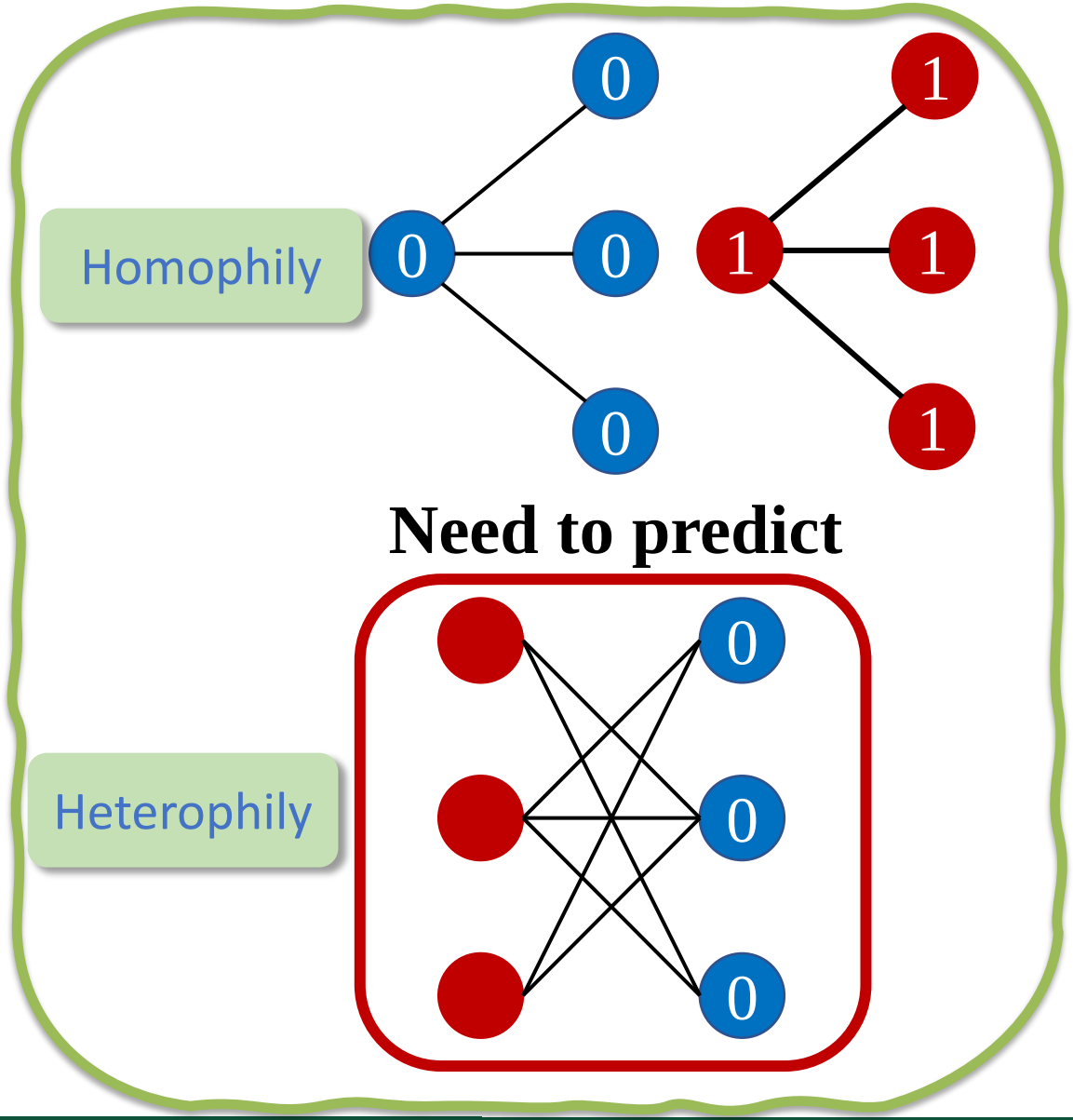
If all the homophily nodes are labeled, all the heterophily nodes are unlabeled



How can GNN work well on homophily & heterophily?

After aggregation

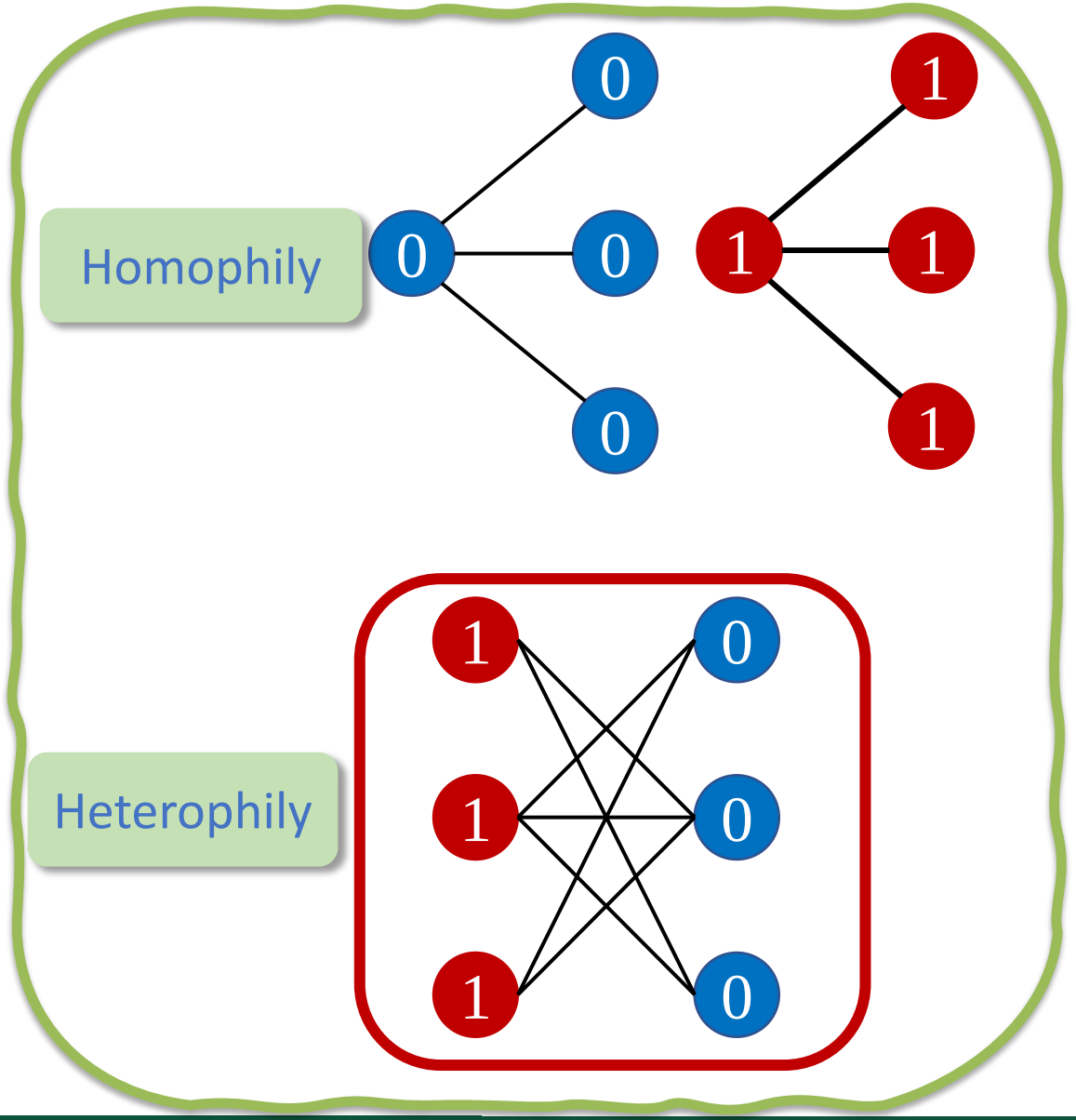
If all the homophily nodes are labeled, all the heterophily nodes are unlabeled



How can GNN work well on homophily & heterophily?

After aggregation

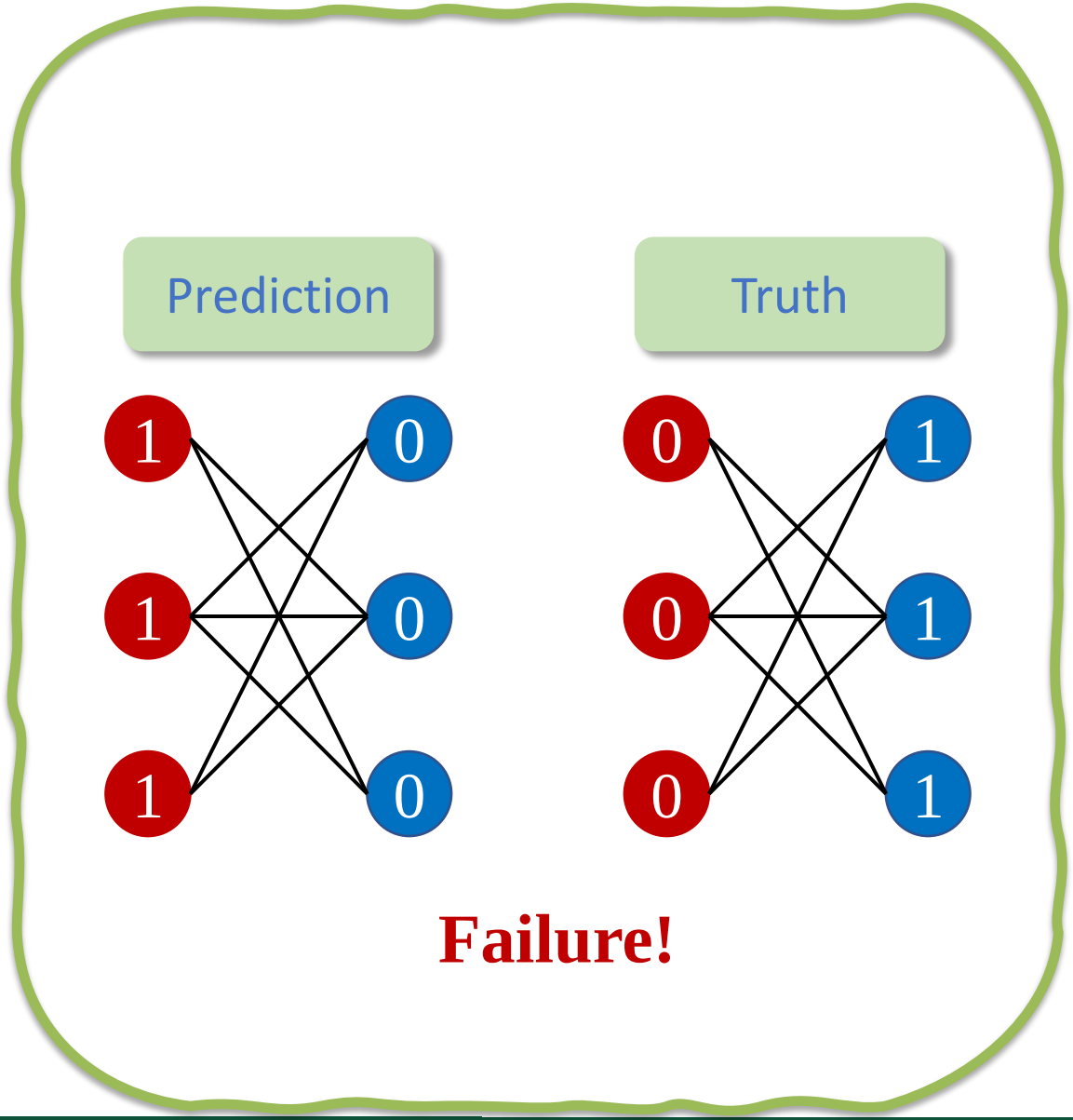
If all the homophily nodes are labeled, all the heterophily nodes are unlabeled



How can GNN work well on homophily & heterophily?

After aggregation

If all homophily nodes are labeled, failures in heterophily nodes.



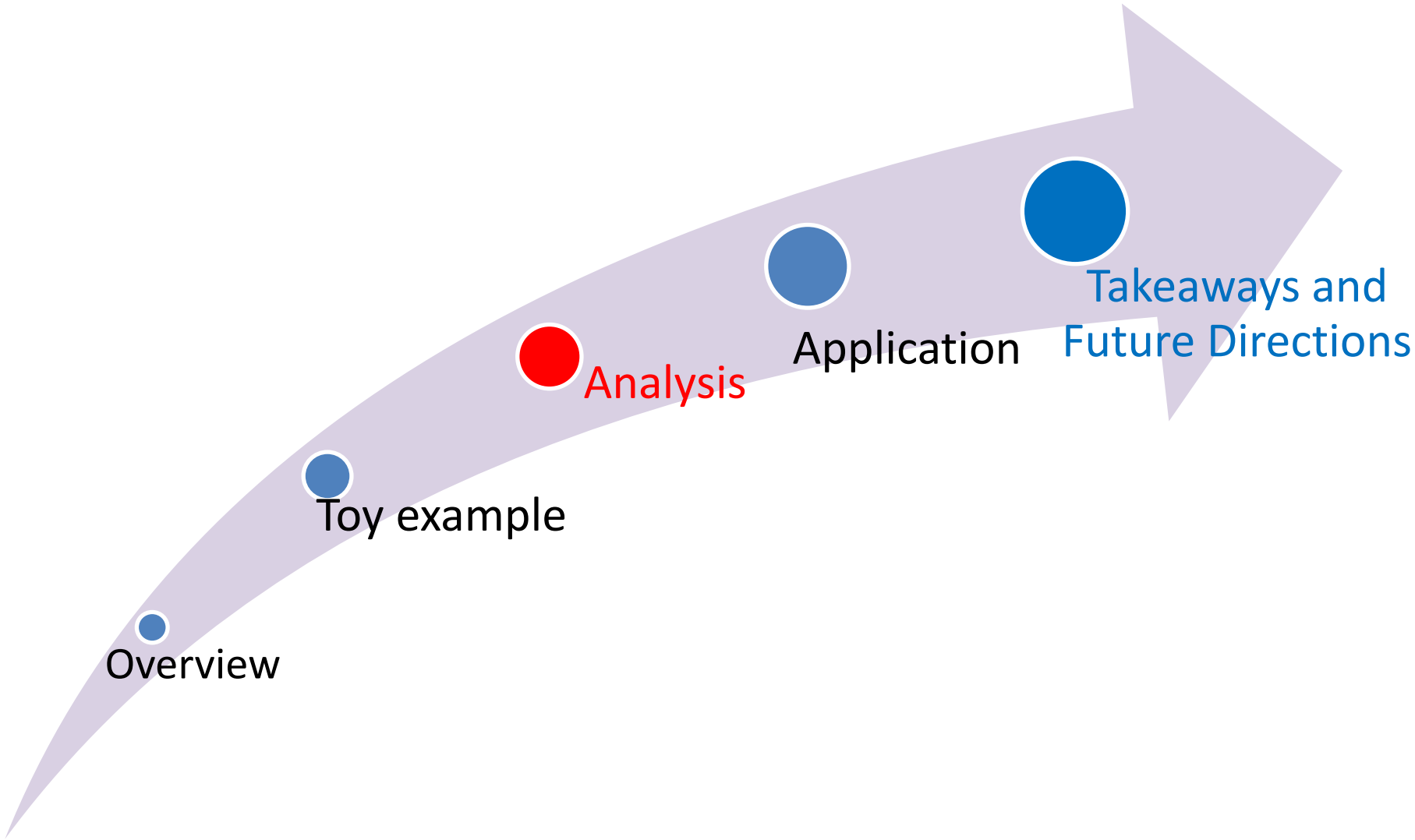
A recap

GNNs help on homophily and heterophily solely.

New behavior when GNNs meet homophily and heterophily nodes together (structure disparity)

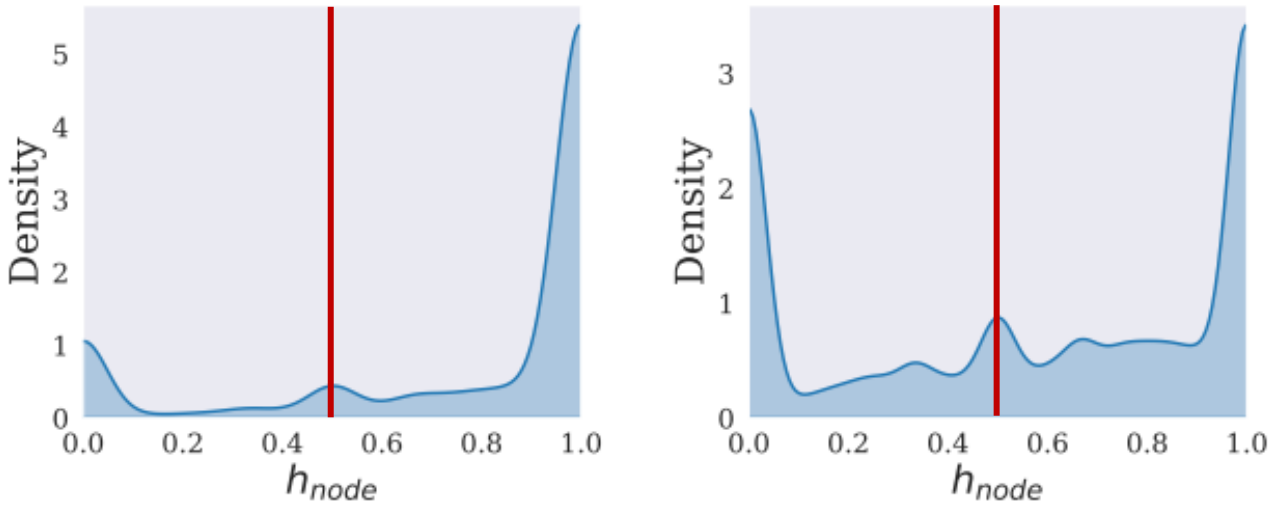
Can one GNN fits all nodes?

Outline


Overview
Toy example
Analysis
Application
Takeaways and
Future Directions

Can one GNN fits all nodes?

Node homophily ratio distribution



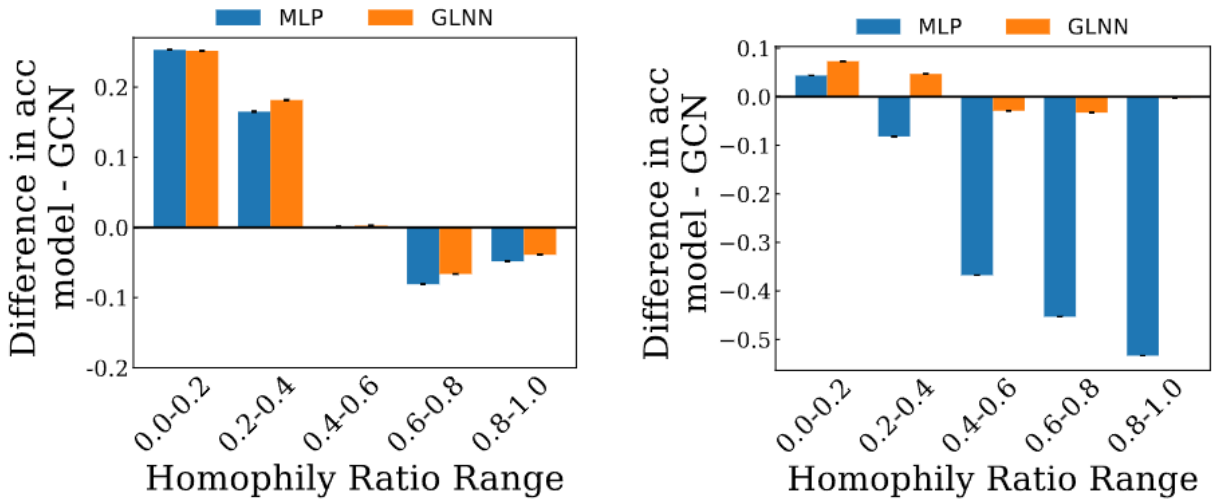
(a) PubMed (h=0.79) (b) Ogbn-arxiv (h=0.63)

Majority pattern : Homophily nodes in a homophily graph

Minority pattern : Heterophily nodes in a homophily graph

Can one GNN fits all nodes?

Performance comparison between GCN and MLP-based models

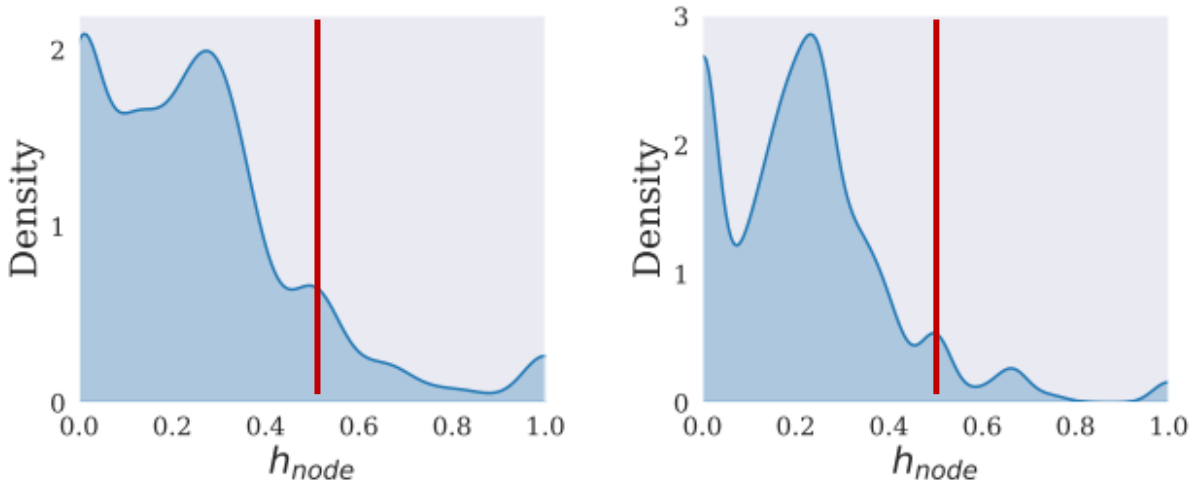


(a) PubMed (h=0.79) (b) Ogbn-arxiv (h=0.63)

GCN outperforms on the majority pattern, but fails in the minor pattern

Can one GNN fits all nodes?

Node homophily ratio distribution



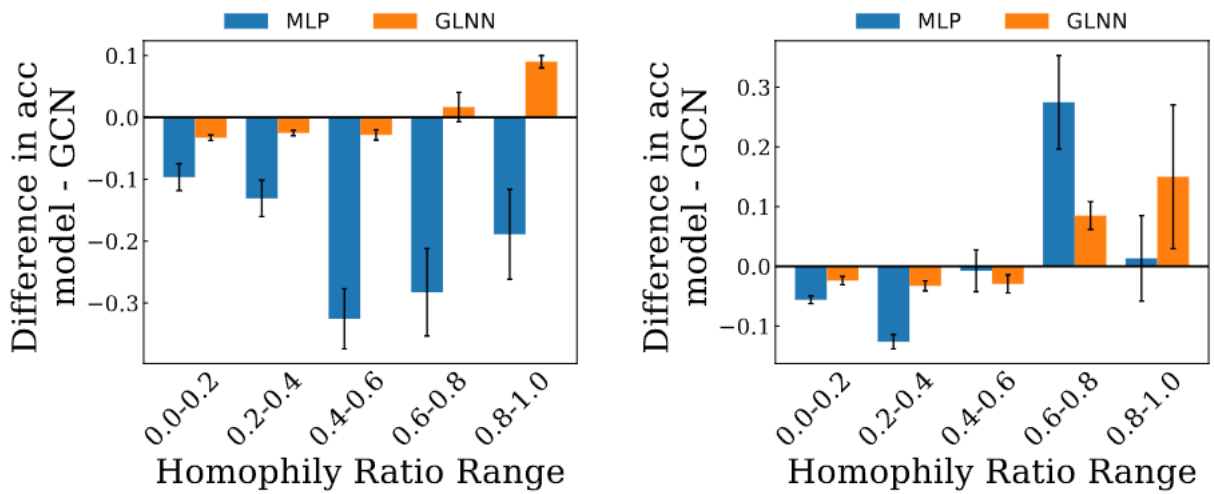
(c) Chameleon (h=0.22) (d) Squirrel (h=0.25)

Majority pattern : Heterophily nodes in a heterophily graph

Minority pattern : Homophily nodes in a heterophily graph

Can one GNN fits all nodes?

Performance comparison between GCN and MLP-based models



(c) Chameleon (h=0.22) (d) Squirrel (h=0.25)

GCN outperforms on the majority pattern, but fails in the minor pattern

Research questions

How Aggregation affects nodes differently?

How Aggregation leads to performance disparity?

Why performance disparity happens on GNN?

How Aggregation affects nodes differently?

Lemma 1. When nodes u and v have the same aggregated features $\mathbf{f}_u = \mathbf{f}_v$ but different structural patterns $h_u \neq h_v$

$$\boxed{|\mathbf{P}_1(y_u = c_1|\mathbf{f}_u) - \mathbf{P}_2(y_v = c_1|\mathbf{f}_v)|} \leq \frac{\rho^2}{\sqrt{2\pi}\sigma}\boxed{|h_u - h_v|}$$

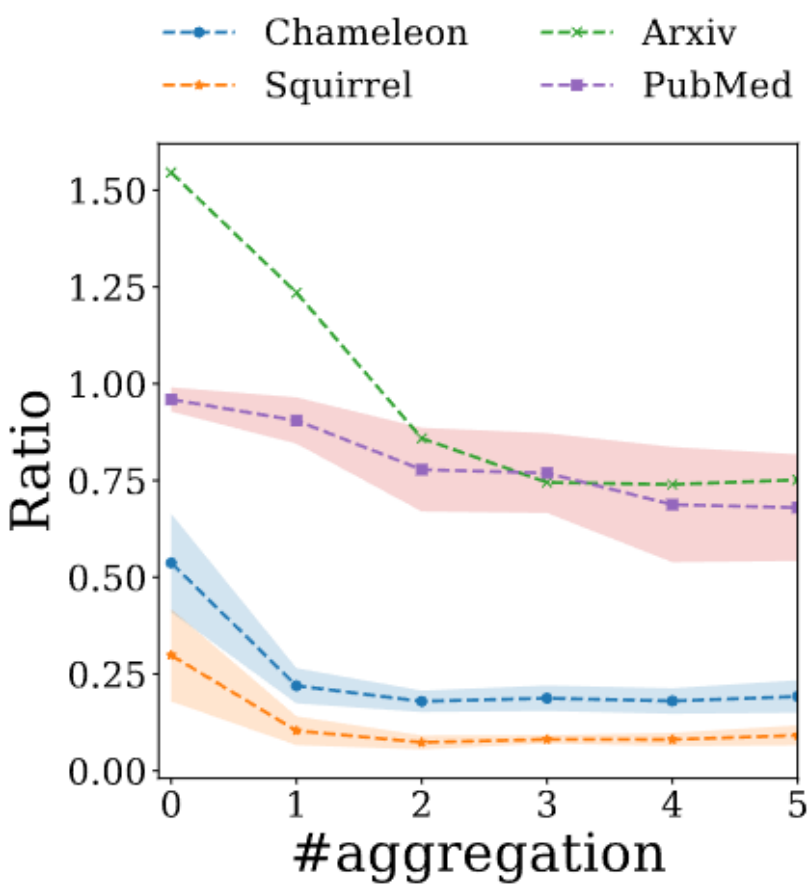
The probability difference on nodes sharing the same class

Homophily ratio Difference (Structure disparity)

Nodes with a small homophily ratio difference are likely to share the same class

How Aggregation leads to performance disparity?

Discriminative ratio variation along with aggregation



Distance between class mean

$$r = \sum_{i=1}^{K} \frac{\boxed{||\boldsymbol{\mu}_i^{\text{tr}} - \boldsymbol{\mu}_i^{\text{ma}}||}}{||\boldsymbol{\mu}_i^{\text{tr}} - \boldsymbol{\mu}_i^{\text{mi}}||}$$

Majority test nodes show better discriminative ability than the minority ones

Why performance disparity happen on GNN?

Theorem 1 (Subgroup Generalization Bound for GNNs). *Let $\tilde{h}$ be any classifier in the classifier family $\mathcal{H}$ with parameters $\{\widetilde{W}_l\}_{l=1}^{L}$. for any $0 < m \leq M$, $\gamma \geq 0$, and large enough number of the training nodes $N_{tr} = |V_{tr}|$, there exist $0 < \alpha < \frac{1}{4}$ with probability at least $1 - \delta$ over the sample of $y^{tr} := \{y_i\}_{i \in V_{tr}}$, we have:*

Train loss

$$\boxed{\mathcal{L}_m^0(\tilde{h})} \leq \boxed{\widehat{\mathcal{L}}_{tr}^{\gamma}(\tilde{h})} + O\left(\underbrace{\frac{K\rho}{\sqrt{2\pi}\sigma}(\epsilon_m + |h_{tr} - h_m| \cdot \rho)}_{(a)} + \underbrace{\frac{b\sum_{l=1}^{L}\|\widetilde{W}_l\|_F^2}{(\gamma/8)^{2/L} N_{tr}^{\alpha}}(\epsilon_m)^{2/L}}_{(b)} + \mathbf{R} \right)$$

Test node subgroup with homophily ratio h_m

Small gap indicates better generalization performance

Why performance disparity happen on GNN?

Theorem 1 (Subgroup Generalization Bound for GNNs). *Let $\tilde{h}$ be any classifier in the classifier family $\mathcal{H}$ with parameters $\{\widetilde{W}_l\}_{l=1}^{L}$. for any $0 < m \leq M$, $\gamma \geq 0$, and large enough number of the training nodes $N_{tr} = |V_{tr}|$, there exist $0 < \alpha < \frac{1}{4}$ with probability at least $1 - \delta$ over the sample of $y^{tr} := \{y_i\}_{i \in V_{tr}}$, we have:*

$$\mathcal{L}_m^0(\tilde{h}) \leq \widehat{\mathcal{L}}_{tr}^{\gamma}(\tilde{h}) + O\left(\underbrace{\frac{K\rho}{\sqrt{2\pi}\sigma}(\epsilon_m + |h_{tr} - h_m| \cdot \rho)}_{(a)} + \underbrace{\frac{b\sum_{l=1}^{L}\|\widetilde{W}_l\|_F^2}{(\gamma/8)^{2/L} N_{tr}^{\alpha}}(\epsilon_m)^{2/L}}_{(b)} + \mathbf{R} \right)$$

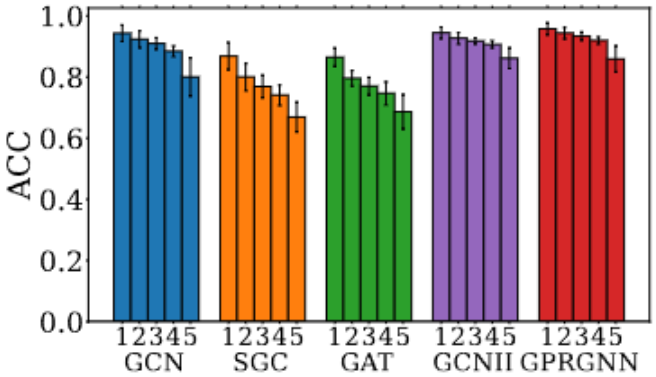
$\epsilon_m = \left|f_i - f_j\right|_F^2$ is the aggregated feature distance between train and test subgroup(s).

$|h_{tr} - h_m|$ is the homophily ratio difference between train and test subgroup(s).

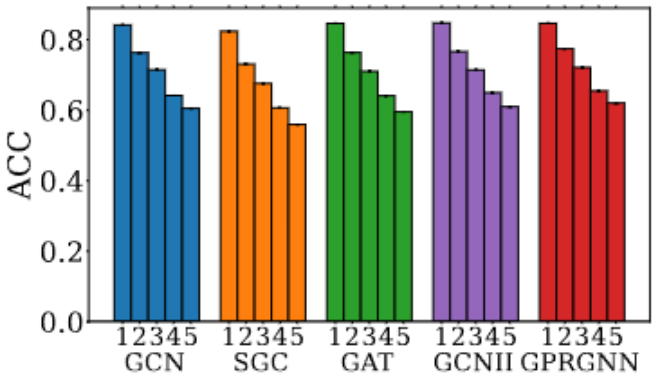
Structure disparity

Empirical verification

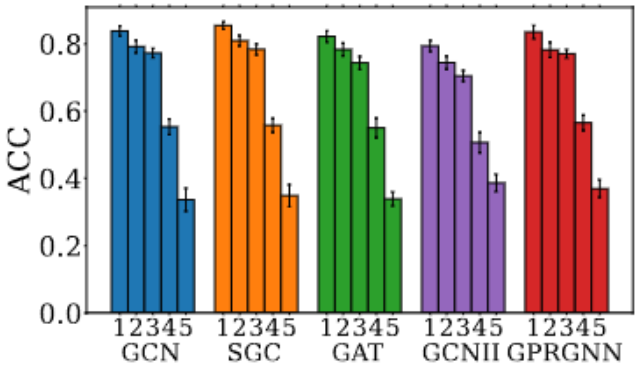
$$s = \epsilon_m + |h_{\text{tr}} - h_m|$$



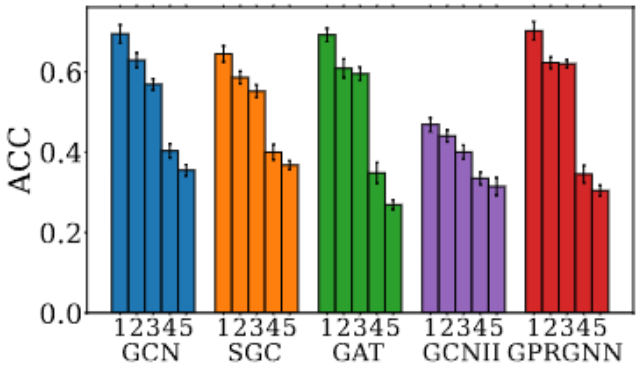
(a) PubMed (h=0.79)



(b) Ogbn-arxiv (h=0.63)



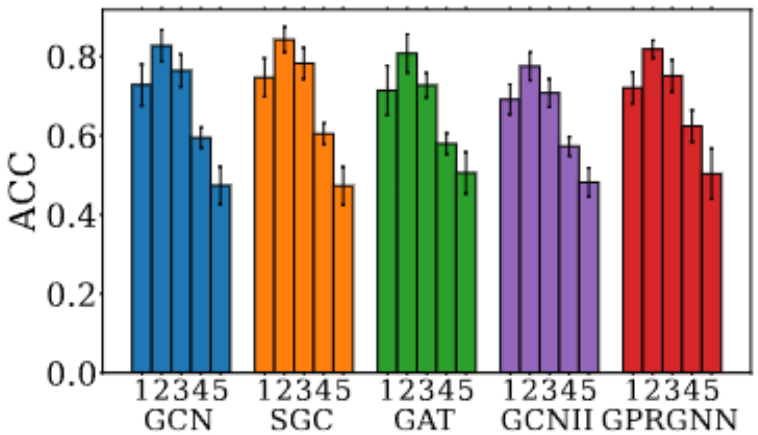
(c) Chameleon (h=0.22)



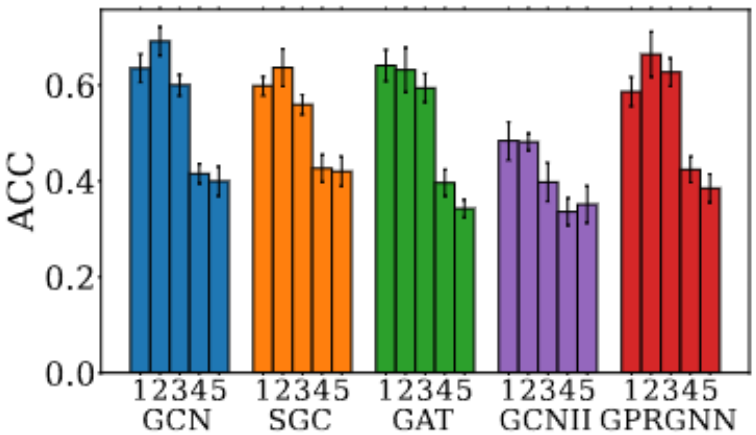
(d) Squirrel (h=0.25)

Empirical verification

$$s = \epsilon_m$$



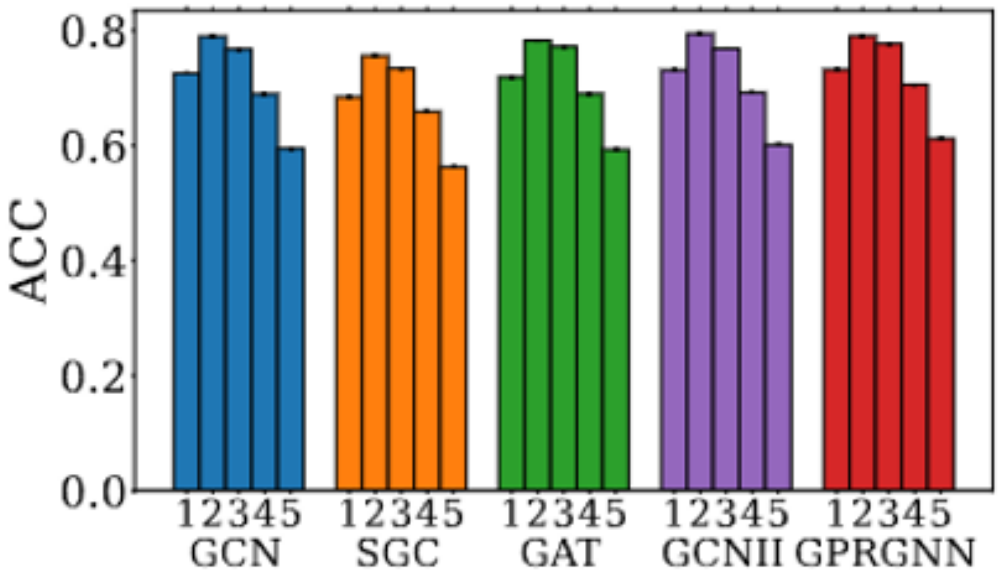
(c) Chameleon (h=0.22)



(d) Squirrel (h=0.25)

Empirical verification

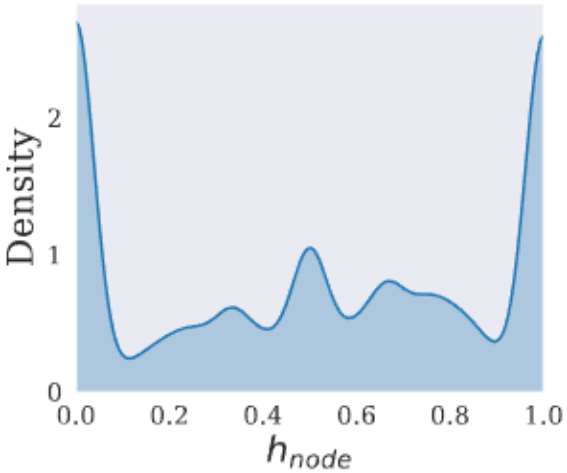
$$s = |h_{\text{tr}} - h_m|$$



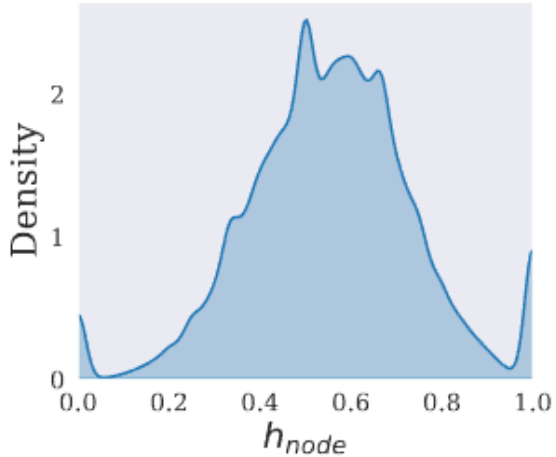
(b) Ogbn-arxiv (h=0.63)

Empirical success on more datasets

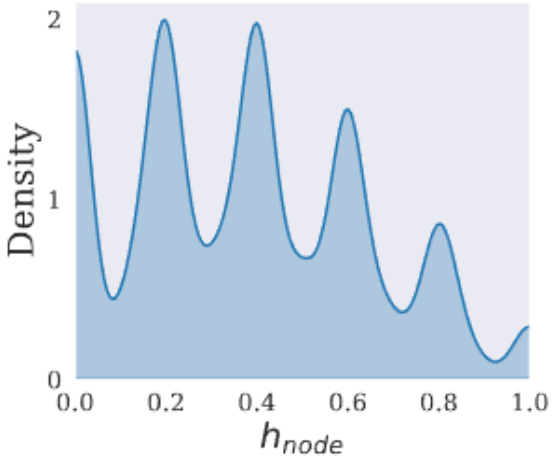
$$s = \epsilon_m + |h_{\text{tr}} - h_m|$$



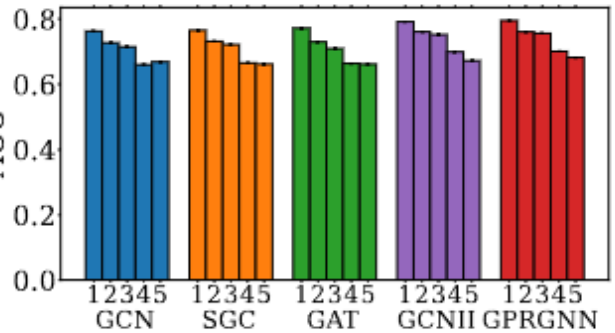
(e) IGB-tiny (h=0.57)



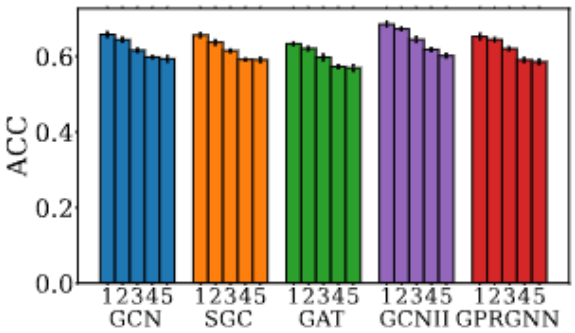
(i) Twitch-gamers (h=0.56)



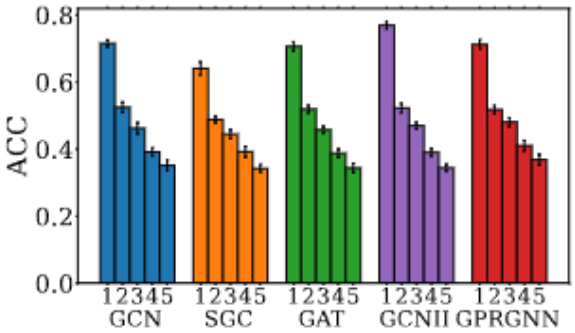
(j) Amazon-ratings (h=0.38)



(c) IGB-Tiny (h=0.57)



(e) Twitch-gamers (h=0.56)



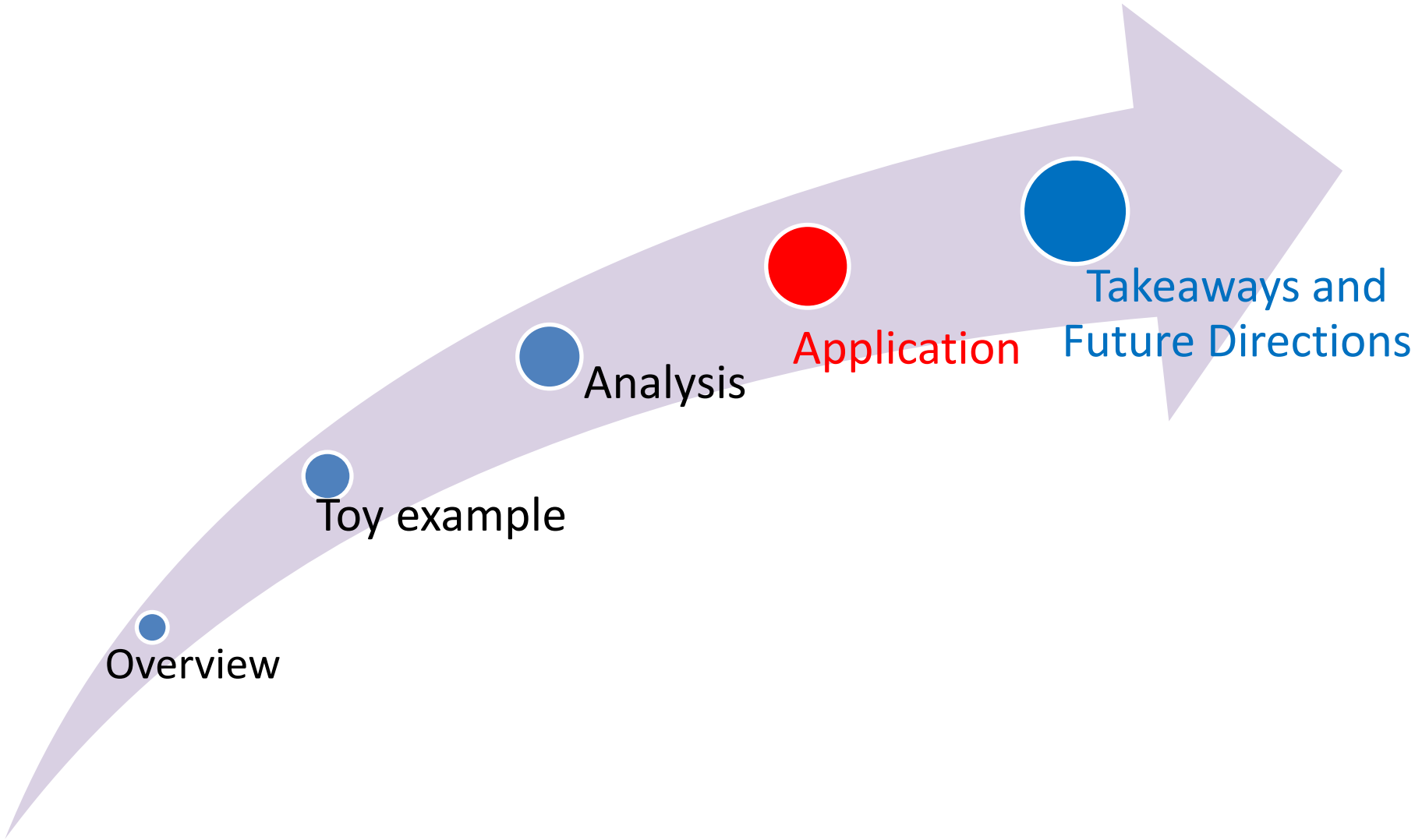
(f) Amazon-ratings (h=0.38)

A recap

Can GNNs learn a good representation for all the nodes?

When can GNNs show good node classification performance?

Outline


Overview
Toy example
Analysis
Application
Takeaways and
Future Directions

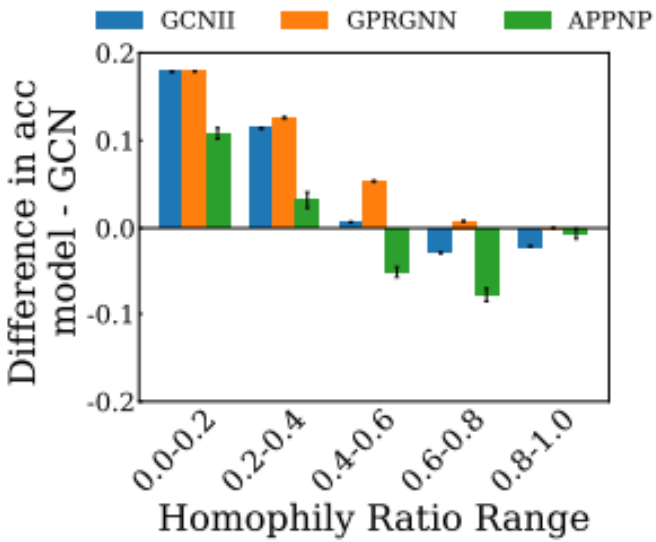
Applications

Elucidating the effectiveness of Deeper GNNs

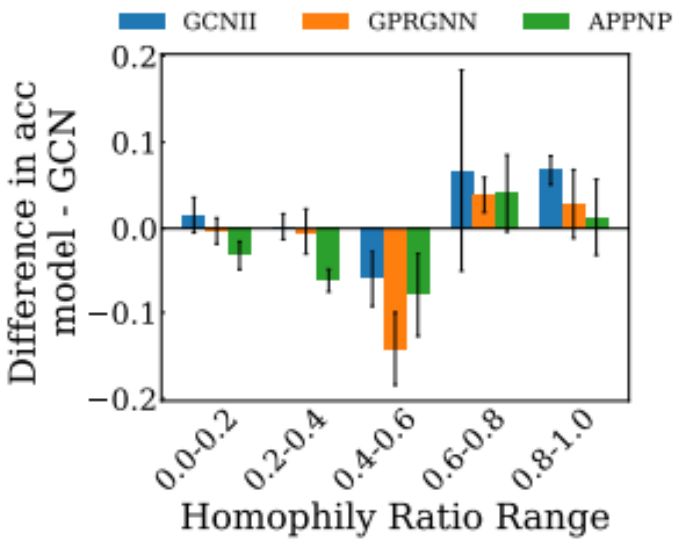
A new practical Graph out-of-distribution scenario

Elucidating the effectiveness of Deeper GNNs

Comparison between GCN and Deeper GNNs



(a) PubMed (h=0.79)



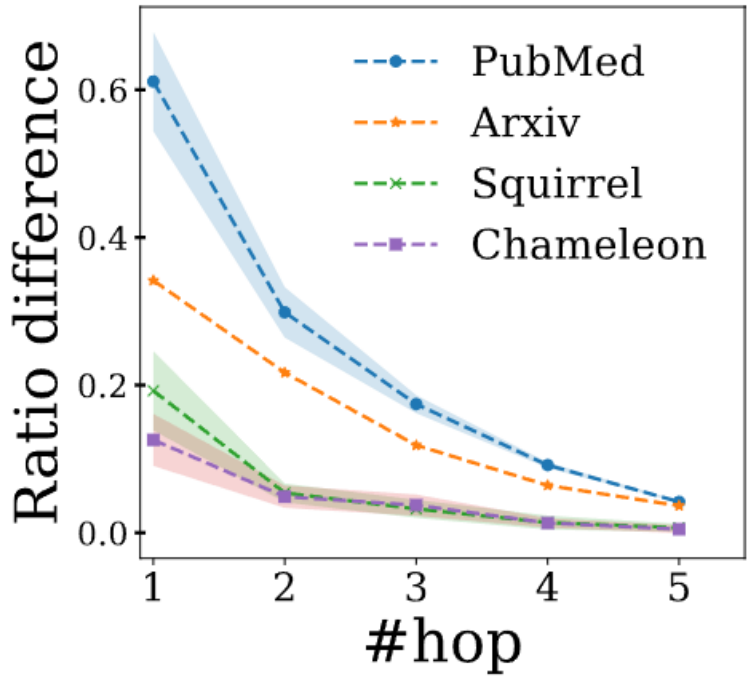
(c) Chameleon (h=0.22)

The performance improvement from Deeper GNNs is from the minority nodes

Elucidating the effectiveness of Deeper GNNs

Multiple-hop homophily ratio differences between training and minority test nodes

$$|h_{\text{tr}} - h_m|$$



The homophily ratio difference on minority pattern decreased in higher-order

A new Graph out-of-distribution scenario

OOD split: majority nodes for train, minority nodes for test

Concept shift

$$P(Y|X_{homo}) \neq P(Y|X_{hete})$$

Not $P(Y|X, e_{train}) \neq P(Y|X, e_{test})$

Hide in existing dataset

Facebook-100

Train Homo	Test Homo
0.18	0.54

ERM	EERM (OOD)
54.04±0.94	54.32±0.60

How Aggregation affects nodes differently?

Lemma 1. When nodes u and v have the same aggregated features $\mathbf{f}_u = \mathbf{f}_v$ but different structural patterns $h_u \neq h_v$

$$\boxed{|\mathbf{P}_1(y_u = c_1|\mathbf{f}_u) - \mathbf{P}_2(y_v = c_1|\mathbf{f}_v)|} \leq \frac{\rho^2}{\sqrt{2\pi}\sigma}\boxed{|h_u - h_v|}$$

The probability difference on nodes sharing the same class

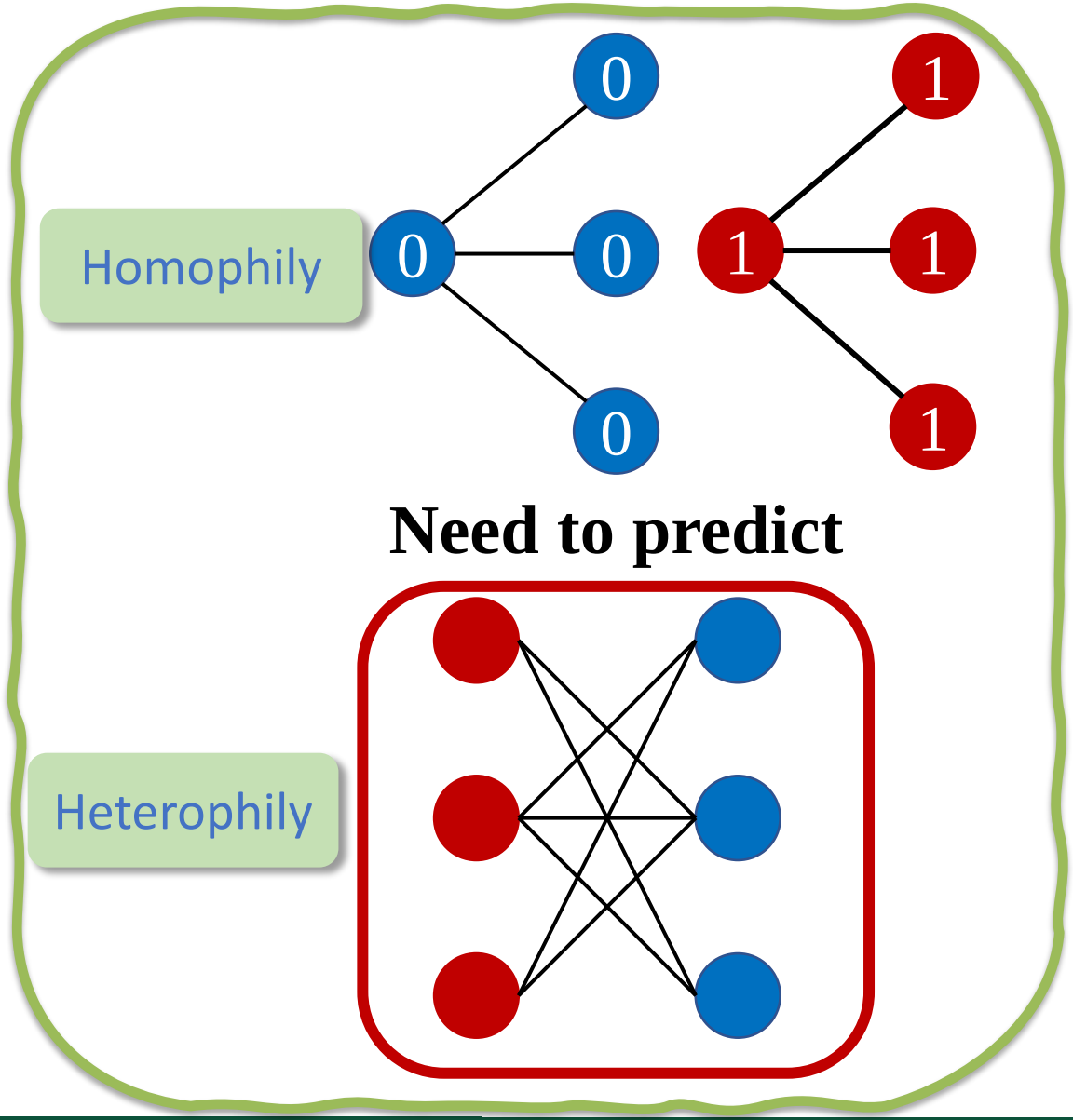
Homophily ratio Difference (Structure disparity)

Nodes with a small homophily ratio difference are likely to share the same class

How can GNN work well on homophily & heterophily?

After aggregation

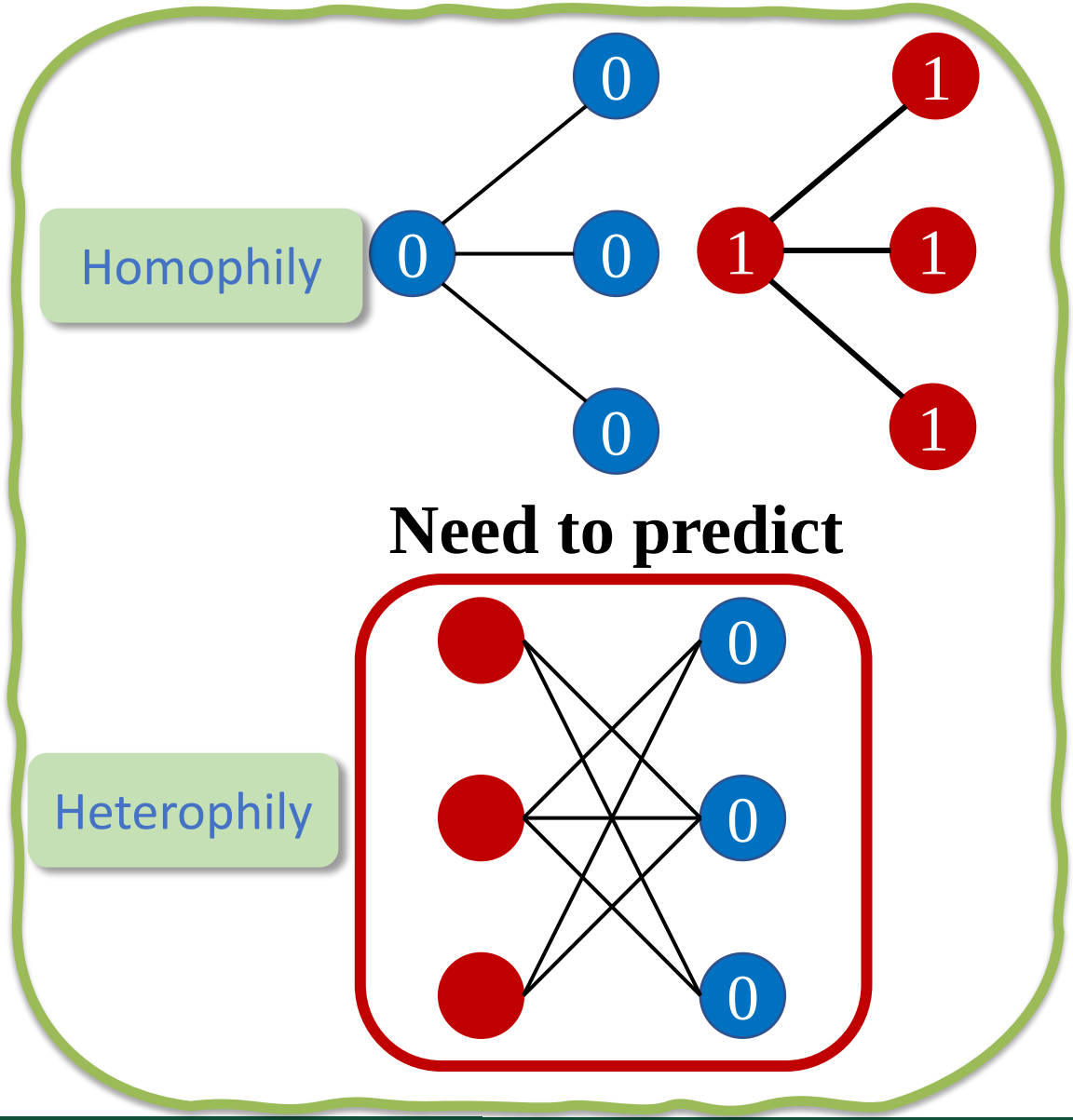
If all the homophily nodes are labeled, all the heterophily nodes are unlabeled



How can GNN work well on homophily & heterophily?

After aggregation

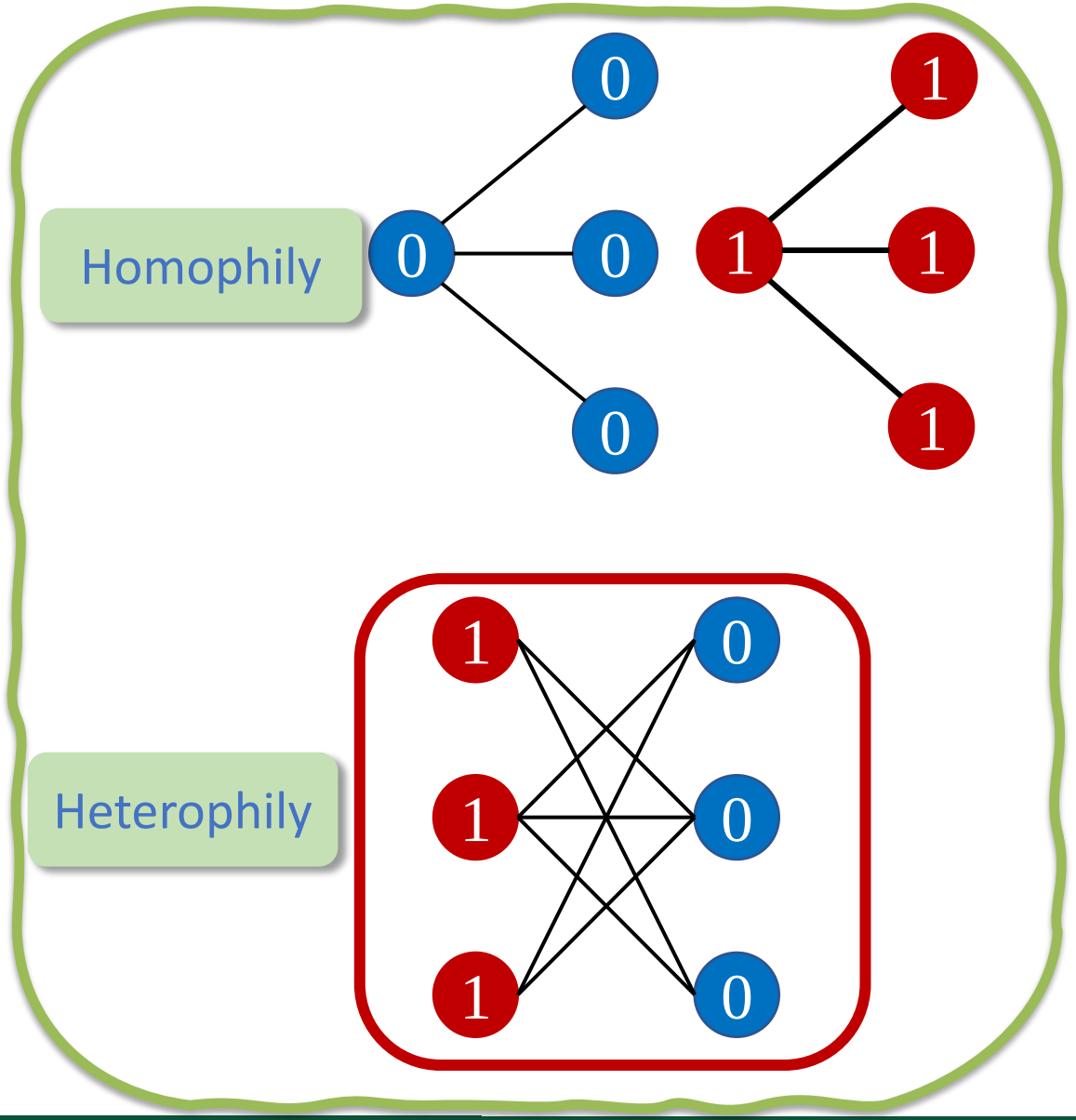
If all the homophily nodes are labeled, all the heterophily nodes are unlabeled



How can GNN work well on homophily & heterophily?

After aggregation

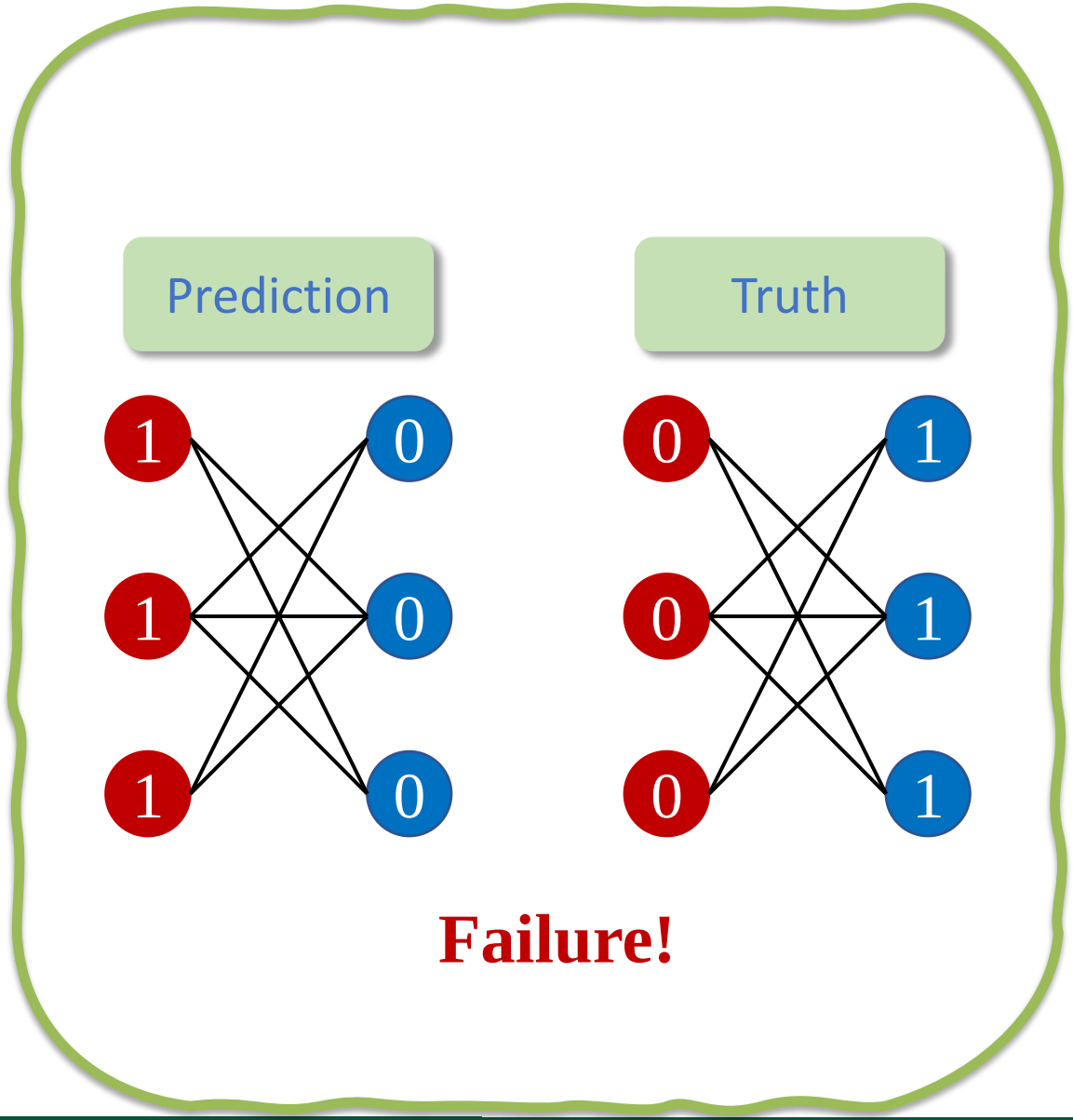
If all the homophily nodes are labeled, all the heterophily nodes are unlabeled



How can GNN work well on homophily & heterophily?

After aggregation

If all homophily nodes are labeled, failures in heterophily nodes.



A new Graph out-of-distribution scenario

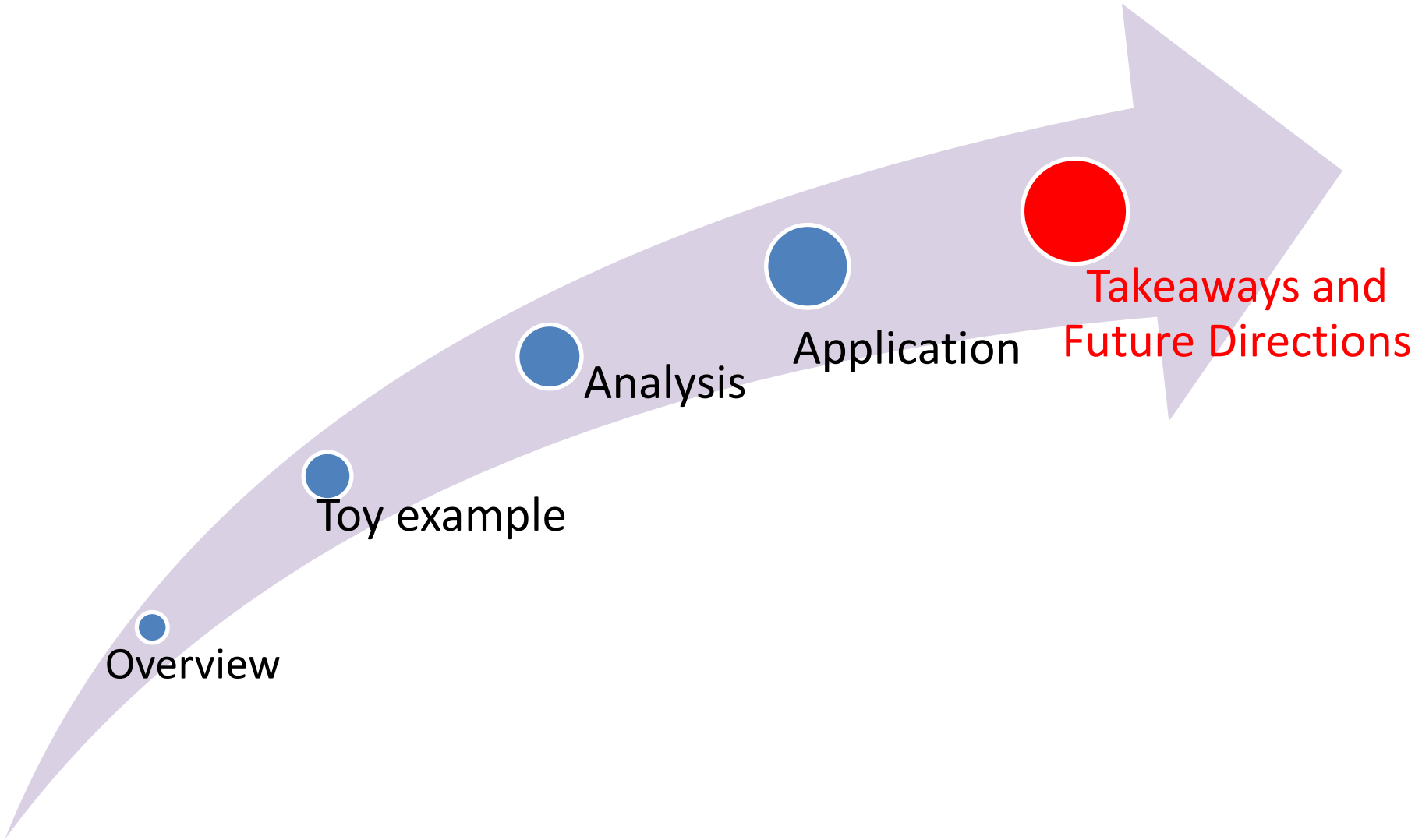
OOD split: majority nodes for train, minority nodes for test

Table 1: Performance (Accuracy) on the proposed OOD split.

	Pubmed	Ogbn-Arxiv	Squirrel	Chameleon
GCN(i.i.d)	89.18±0.15	72.99±0.14	58.09±0.71	75.09±0.79
GCN	51.04±0.16	34.94±0.07	32.13±4.93	43.35±3.47
MLP	68.38±0.43	33.17±0.37	24.57±0.77	34.78±4.97
GLNN	67.51±0.25	35.89±0.14	31.51±0.70	47.01±1.09
GCNII	67.76±0.36	36.81±0.14	37.15±1.39	41.25±2.03
GPRGNN	57.24±0.18	34.95±0.43	42.43±7.71	35.27±7.67
SRGNN	57.91±0.10	40.37±1.65	37.62±1.74	42.09±0.43
EERM	65.37±1.35	34.23±0.46	40.93±0.57	45.84±1.05
EERM(II)	67.59±0.91	40.28±0.84	44.31±0.40	48.59±0.78

OOD methods do not work well

Outline


Overview
Toy example
Analysis
Application
Takeaways and
Future Directions

Main Takeaways

Reveals the inner workings of GNNs

Identifies the limitations of GNNs across datasets

Inspire new principled applications on GNNs

Future work

Build effective solutions for the proposed OOD issue

More understandings with higher-order homophily ratio

Build robust GNNs without structural disparity issue

Align with understandings on Graph Robust and OOD

GNNs: look Ahead

Build toy example to describe graphs on more task

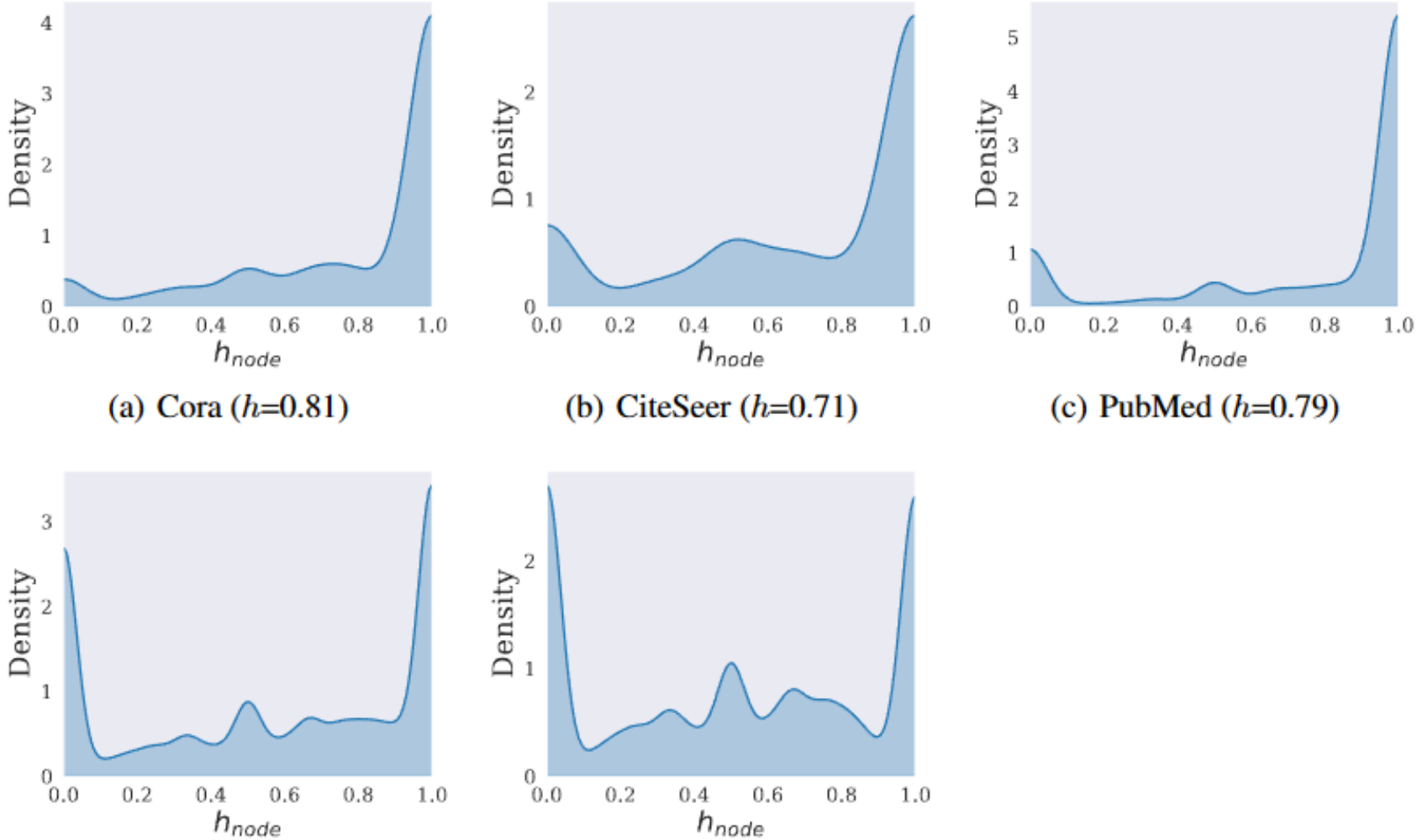
Understand the key factors underlying GNN success

Find exact scenarios when and why GNN can and cannot work

Thanks & QA!

Appendix

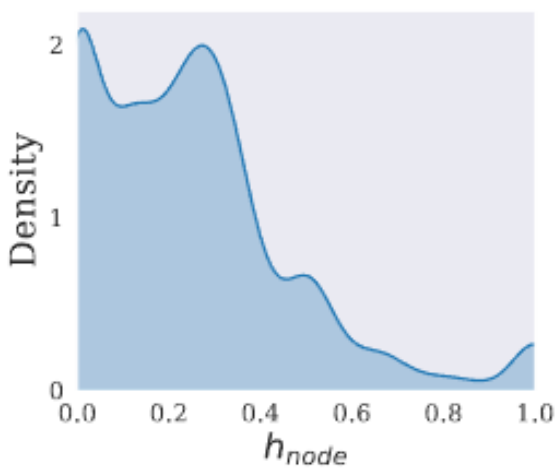
Homophily ratio distribution



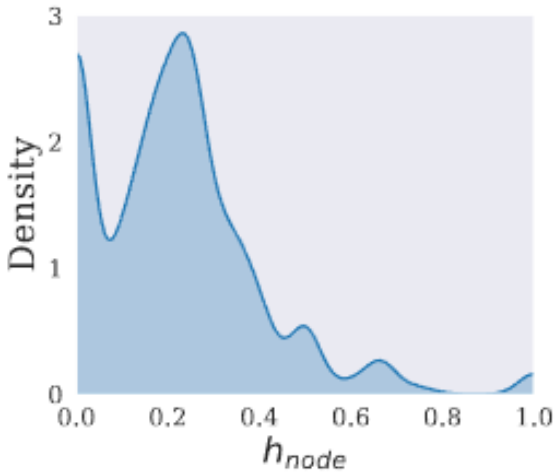
(a) Cora (h=0.81) (b) CiteSeer (h=0.71) (c) PubMed (h=0.79)

(d) Ogbn-Arxiv (h=0.63) (e) IGB-tiny (h=0.57)

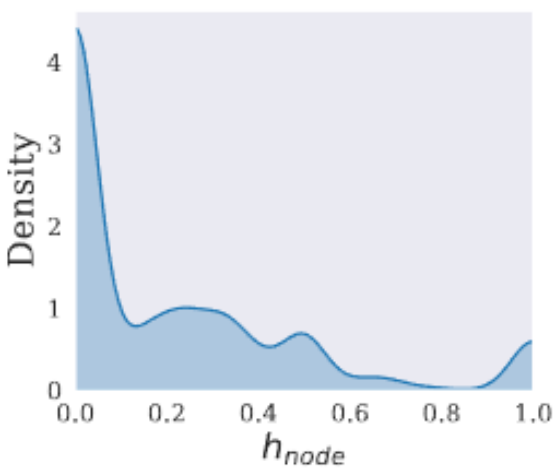
Homophily ratio distribution



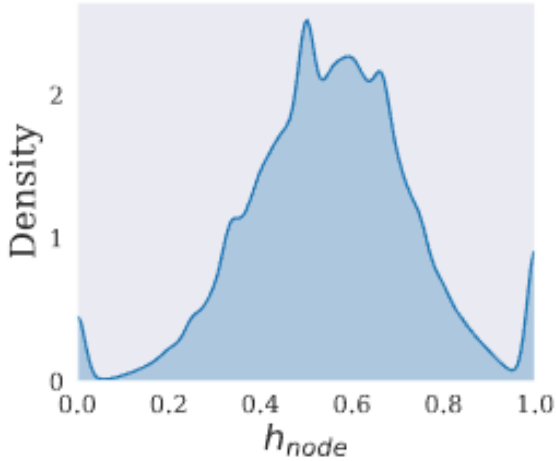
(f) Chameleon (h=0.25)



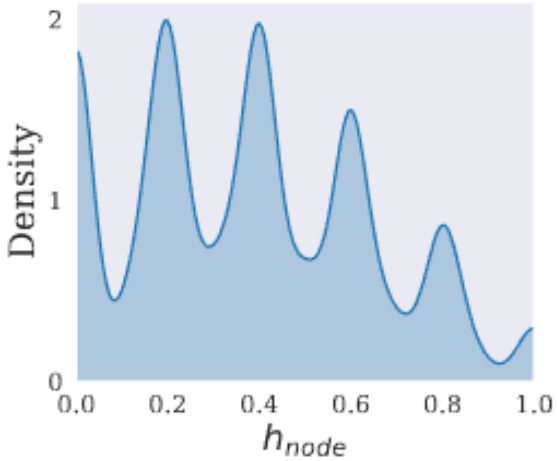
(g) Squirrel (h=0.22)



(h) Actor (h=0.22)



(i) Twitch-gamers (h=0.56)



(j) Amazon-ratings (h=0.38)

Model performance

Table 10: The accuracy of GNN and MLP models on homophilic graphs

Dataset	Cora	Citeseer	Pubmed	Arxiv	IGB-tiny
MLP	61.1±1.2	60.0±1.4	69.0±2.3	54.0±0.1	73.2±0.1
GLNN	81.3±1.5	73.0±2.7	78.2±2.6	71.7±0.1	73.2±0.1
GCN	81.5±1.4	73.7±1.6	77.9±2.0	71.4±0.1	70.7±0.1
SGC	81.7±1.4	72.7±2.2	77.0±2.7	68.0±0.1	71.0±0.1
GAT	82.2±1.1	73.6±1.6	77.3±1.5	71.0±0.1	70.8±0.2
APPNP	83.1±1.3	75.0±1.1	79.6±1.3	70.3±0.5	71.2±0.1
GCNII	82.8±1.1	73.8±1.7	79.0±2.5	71.7±0.5	73.5±0.1
GPRGNN	82.9±1.4	72.4±1.8	78.3±2.1	72.3±0.3	73.9±0.1

Table 11: The accuracy of GNN and MLP models on heterophilic graphs

Dataset	Chameleon	Squirrel	Twitch-gamers	Actor	Amazon-ratings
MLP	49.0±2.4	30.1±1.7	60.7±0.2	37.0±0.7	45.9±0.8
GLNN	39.2±2.7	52.3±1.4	61.1±0.1	37.3±1.0	54.0±0.7
GCN	68.0±2.0	54.7±1.4	62.2±0.2	30.7±0.9	49.0±0.6
SGC	69.1±1.8	53.0±1.1	62.0±2.0	30.0±1.5	46.5±0.6
GAT	67.0±1.9	53.2±1.7	59.9±0.3	30.7±1.0	48.0±0.5
APPNP	56.7±2.5	42.4±1.9	59.7±0.1	37.0±1.3	44.9±0.8
GCNII	64.7±1.8	44.0±1.5	64.5±0.3	36.0±1.2	50.0±0.5
GPRGNN	68.5±1.4	53.8±1.4	61.9±0.2	36.5±1.4	49.8±0.5

Data and model Assumption

Definition 1 (CSBM-S$(\mu_1, \mu_2, (p^{(1)}, q^{(1)}), (p^{(2)}, q^{(2)}), \Pr(\text{homo})))$. *The generated nodes consist of two disjoint sets $\mathcal{C}_1$ and $\mathcal{C}_2$. each node feature x is sampled from $N(\mu_i, I)$ with $i \in \{1, 2\}$. Each set $\mathcal{C}_i$ consists of two subgroups: $\mathcal{C}_i^{(1)}$ for nodes in homophilic pattern with intra-class and inter-class edge probability $p^{(1)} > q^{(1)}$ and $\mathcal{C}_i^{(2)}$ for nodes in heterophilic pattern with $p^{(2)} > q^{(2)}$. $\Pr(homo)$ denotes the probability that the node is in homophilic pattern. $\mathcal{C}_i^{(j)}$ denotes node in class i and subgroup j with $(p^{(j)}, q^{(j)})$. We assume nodes follow the same degree distribution with $p^{(1)} + q^{(1)} = p^{(2)} + q^{(2)}$.*

Definition 2 (Generalized CSBM-S model). *Each node subgroup V_m follows the CSBM distribution $V_m \sim CSBM(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, p^{(i)}, q^{(i)})$, where different subgroups share the same class mean but different intra-class and inter-class probabilities $p^{(i)}$ and $q^{(i)}$. Moreover, node subgroups also share the same degree distribution as $p^{(i)} + q^{(i)} = p^{(j)} + q^{(j)}$.*

Assumption 1 (GNN model). *We focus on SGC [15] with the following components: (1) a one-hop mean aggregation function g with $g(X, G)$ denoting the output. (2) MLP feature transformation $f(g_i(X, G); W_1, W_2, \cdots, W_L)$, where f is a ReLU-activated L-layer MLP with $W_1, \cdots, W_L$ as parameters for each layer. The largest width of all the hidden layers is denoted as b.*

Addition theoretical analysis on linear seperability

Lemma 2 (Linear separability on nodes with the same structural patterns). *Considering mean aggregated features are from the same structural pattern* $\mathbf{f}_i^{(j)}$, *for* $i \in \{1, 2\}$. *For any node* i, *the largest-margin linear classifier on* $\mathbf{f}_i^{(j)}$ *will have a lower probability to misclassify than* $\mathbf{x}_i$, *when* $d_i > \frac{(p^{(1)}+q^{(1)})^2}{(p^{(1)}-q^{(1)})^2}$

When $p^{(1)} = 0.9, q^{(1)} = 0.1$:

Improved linear separability can be found with $d_i > 1.75$

Lemma 3 (Linear separability on nodes with different structural patterns). *Consider features are from different structural patterns, where* $\mathbf{f}_i^{(1)}$ *for* $i \in \mathcal{C}_1$ *and* $\mathbf{f}_i^{(2)}$ *for* $i \in \mathcal{C}_2$. *For any node* i, *the largest-margin linear classifier will have a lower probability to misclassify* $\mathbf{f}_i^{(1)}$ *for* $i \in \mathcal{C}_1$ *and* $\mathbf{f}_i^{(2)}$ *for* $i \in \mathcal{C}_2$ *than* $\mathbf{x}_i$ *when* $d_i > \frac{(p^{(1)}+q^{(1)})^2}{(p^{(1)}-q^{(2)})^2}$

When $p^{(2)} = 0.2, q^{(2)} = 0.8$,

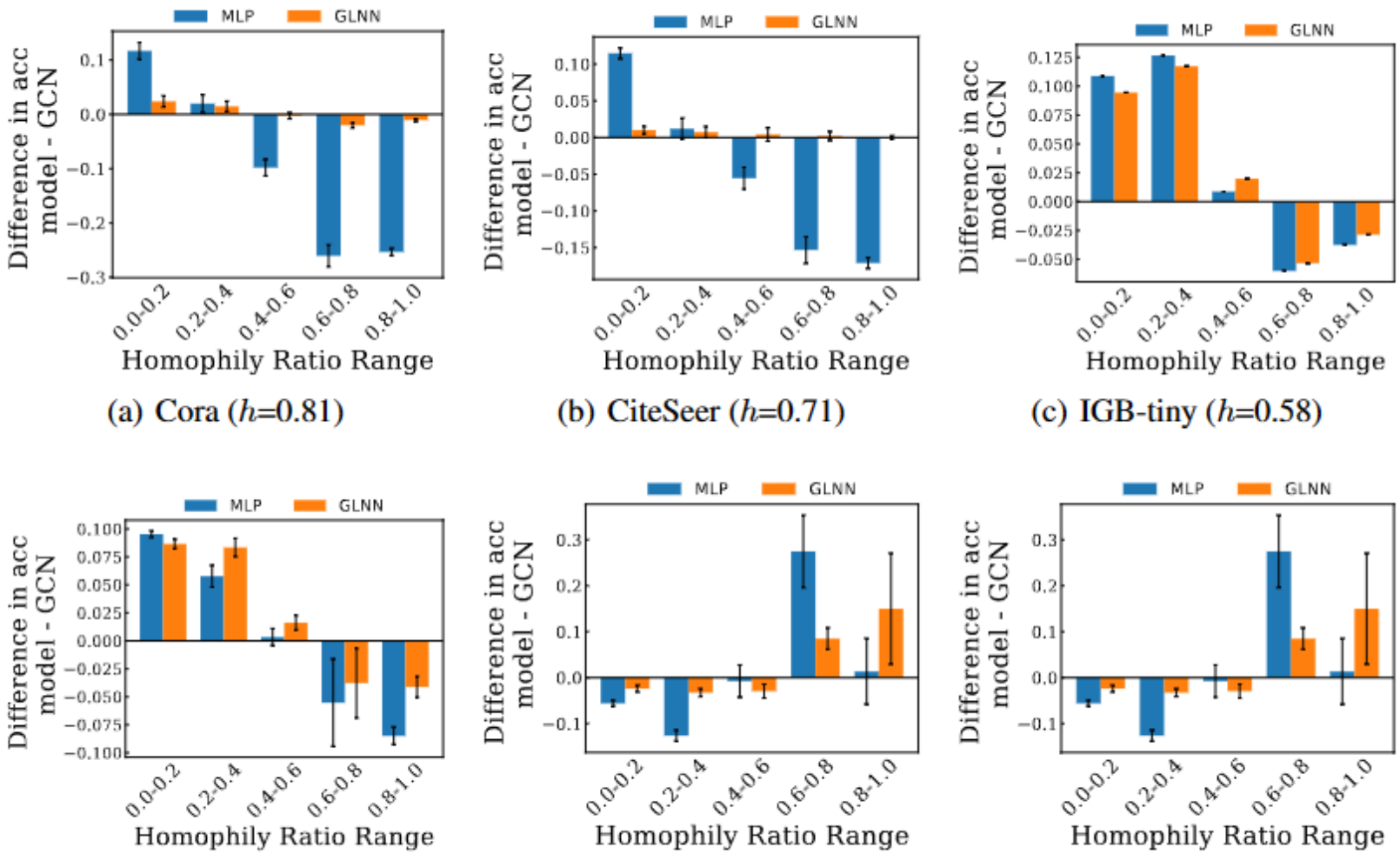
Improved linear separability can be found with $d_i > 100$

Empirical verfication

Table 6: The performance of logistic regression algorithm on homophilic nodes, heterophilic nodes, and a mixture of homophilic and heterophilic nodes. The results on the first row and first column correpond to the performance on homophilic nodes and heterophilic nodes, solely.

Hete\Homo	-	p=0.01, q=0.005	p=0.01, q=0.003	p=0.01, q=0.001
-	-	74.68±3.19	82.71±1.86	92.08±1.13
p=0.001, q=0.005	79.64±2.11	60.84±0.64	62.08±0.59	81.38±1.02
p=0.001, q=0.003	70.08±1.71	59.72±2.01	61.58±1.08	76.60±0.98
p=0.001, q=0.002	62.08±3.04	65.92±1.95	69.42±1.03	74.16±1.09

Additional comparison between GCN and MLP



(a) Cora (h=0.81) (b) CiteSeer (h=0.71) (c) IGB-tiny (h=0.58)

(d) Actor (h=0.22) (e) twitch-gamers (h=0.56) (f) Amazon-ratings (h=0.38)

Additional synthetic analysis

nodes with low homophily

Adding K intra-class edges

nodes with high homophily

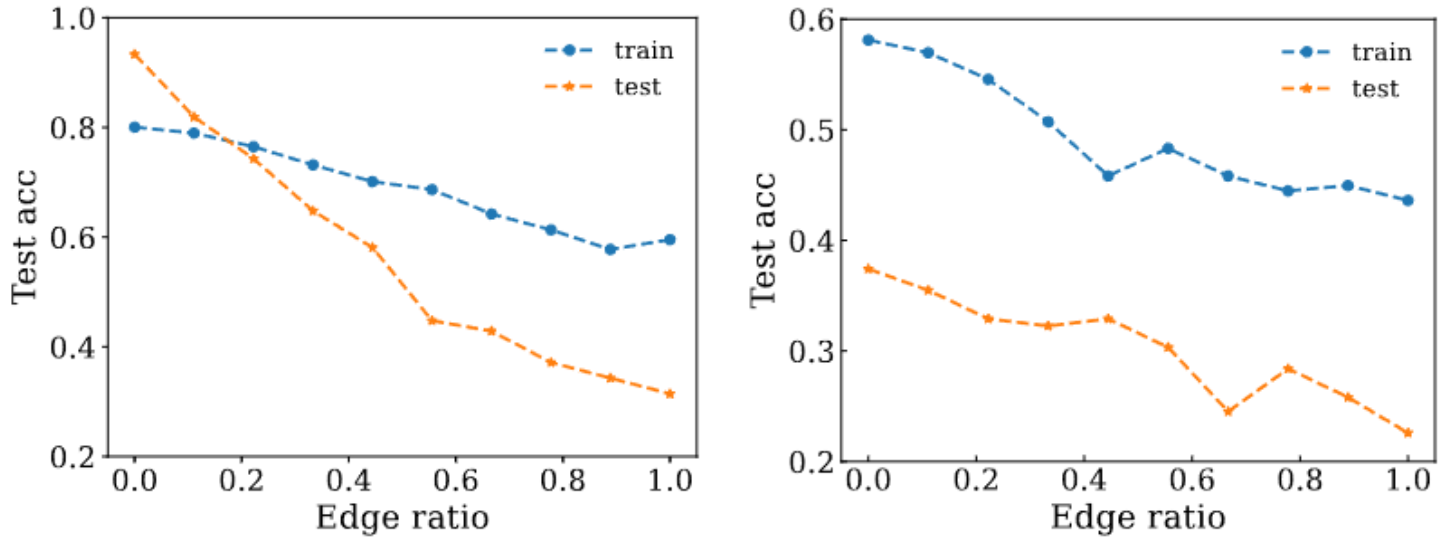
$$\mathcal{D}_0 : \text{Categorical}([0, 0.5, 0, 0, 0, 0.5]),$$
$$\mathcal{D}_1 : \text{Categorical}([0.5, 0, 0.5, 0, 0, 0]),$$
$$\mathcal{D}_2 : \text{Categorical}([0, 0.5, 0, 0.5, 0, 0]),$$
$$\mathcal{D}_3 : \text{Categorical}([0, 0, 0.5, 0, 0.5, 0]),$$
$$\mathcal{D}_4 : \text{Categorical}([0, 0, 0, 0.5, 0, 0.5]),$$
$$\mathcal{D}_5 : \text{Categorical}([0.5, 0, 0, 0, 0.5, 0]).$$

nodes with high homophily

Adding K across-class edges

nodes with low homophily

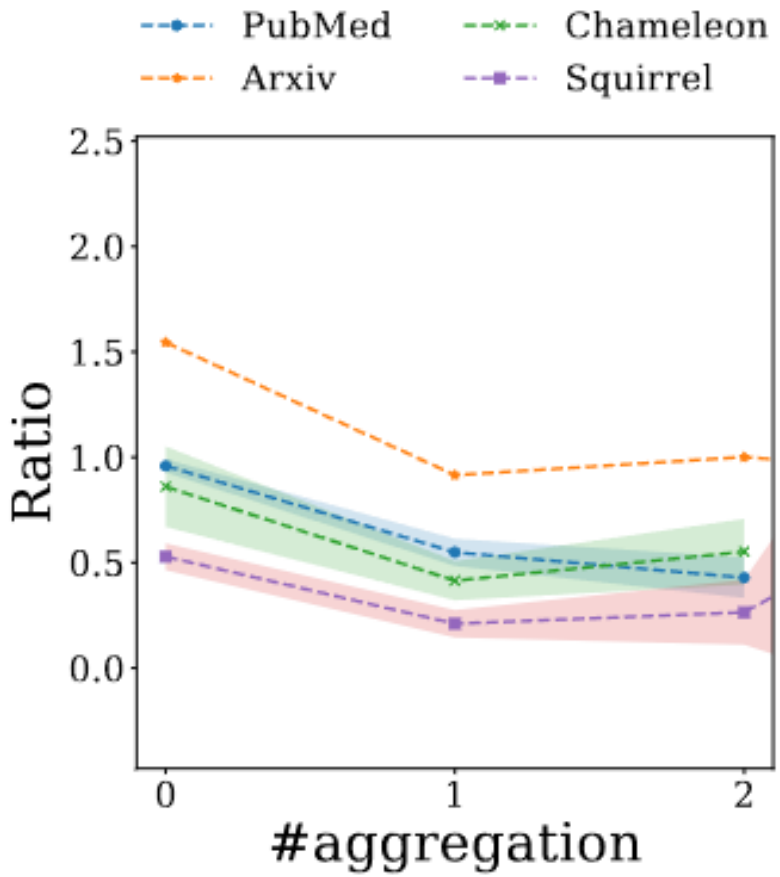
Additional synthetic analysis



(a) Synthetic graphs generated from Cora with the targeted heterophilic edge algorithm

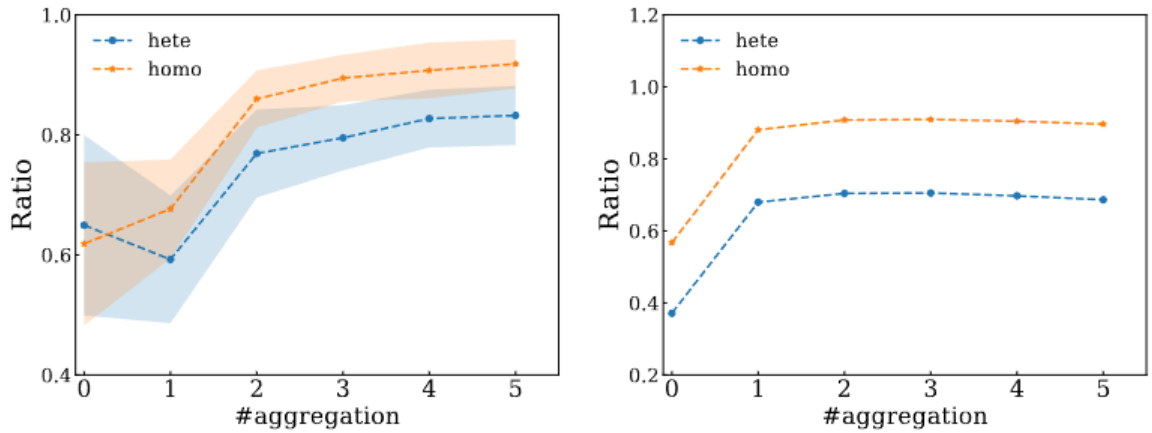
(b) Synthetic graphs generated from Squirrel with the targeted homophilious edge algorithm

Additional discriminative analysis on GCN



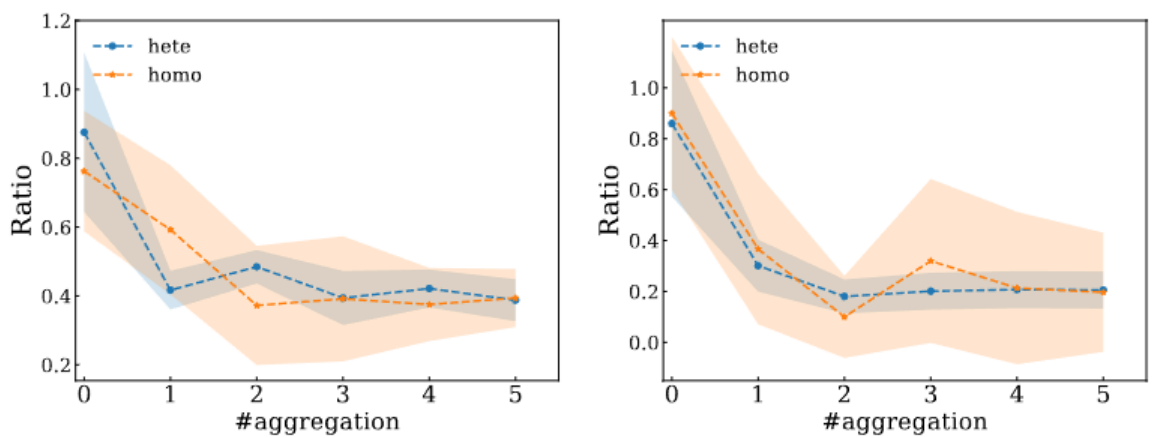
Additional local discriminative analysis

$$r = \frac{\sum_{v \in V_{\text{te}}} \mathbb{1}\left[\exists c \in \mathcal{C}, |\mathcal{M}_v^c| > \frac{k}{2}\right]}{|V_{\text{te}}|}$$



(c) PubMed (h=0.79)

(d) Ogbn-arxiv (h=0.63)

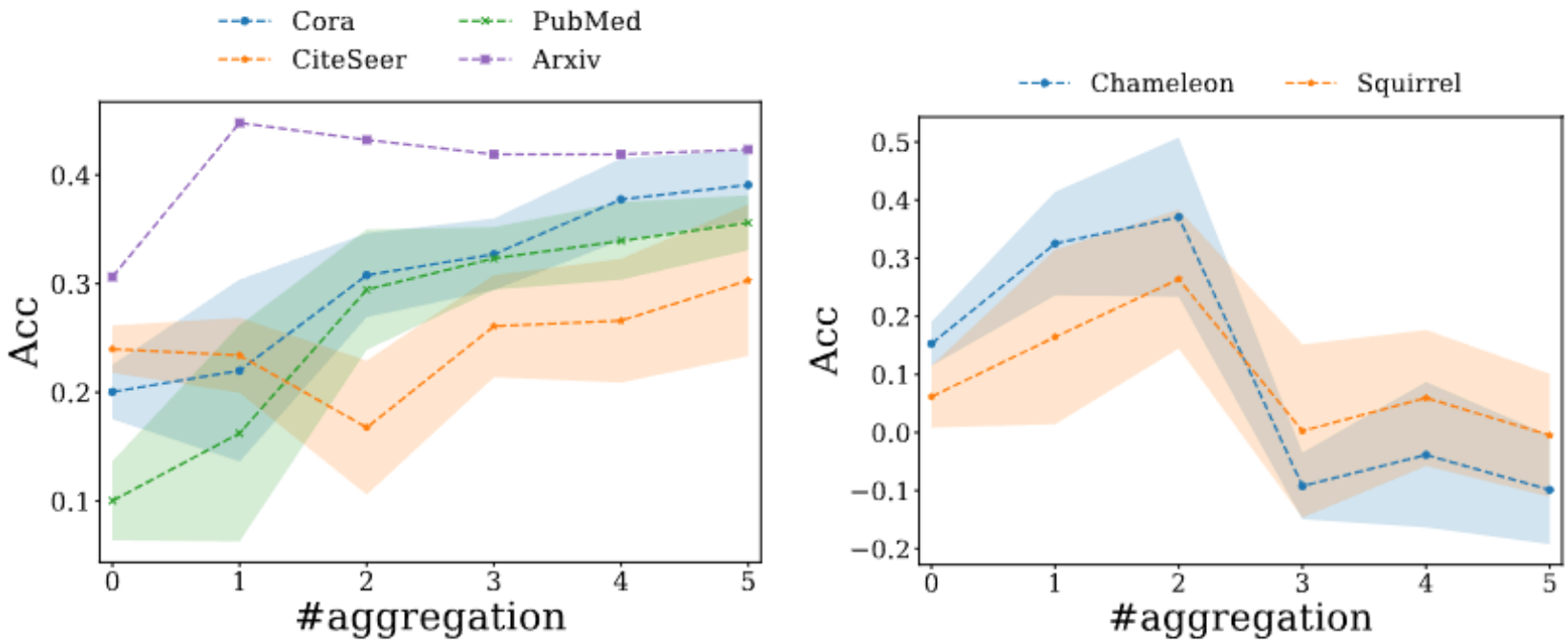


(e) Chameleon (h=0.25)

(f) Squirrel (h=0.22)

Additional local discriminative analysis

$$\mathrm{Acc}_{\mathrm{local}} = \frac{\sum_{v \in V_{\mathrm{agree}}} \mathbb{1}\left[c_v = c_{\mathcal{N}_v}\right]}{|V_{\mathrm{agree}}|}$$

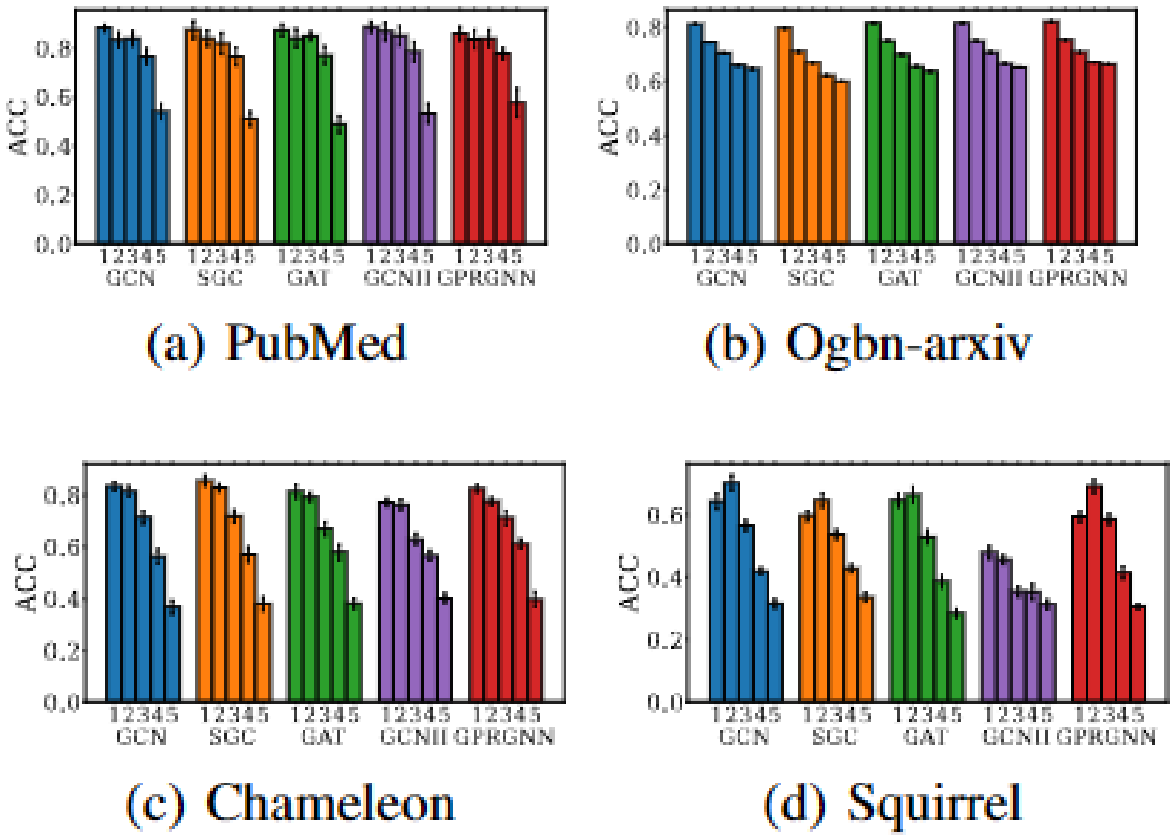


(a) Homophilic graphs

(b) Heterophilic graphs

Additional higher-order performance disparity

$$s = \epsilon_m + |h_{\text{tr}} - h_m|$$

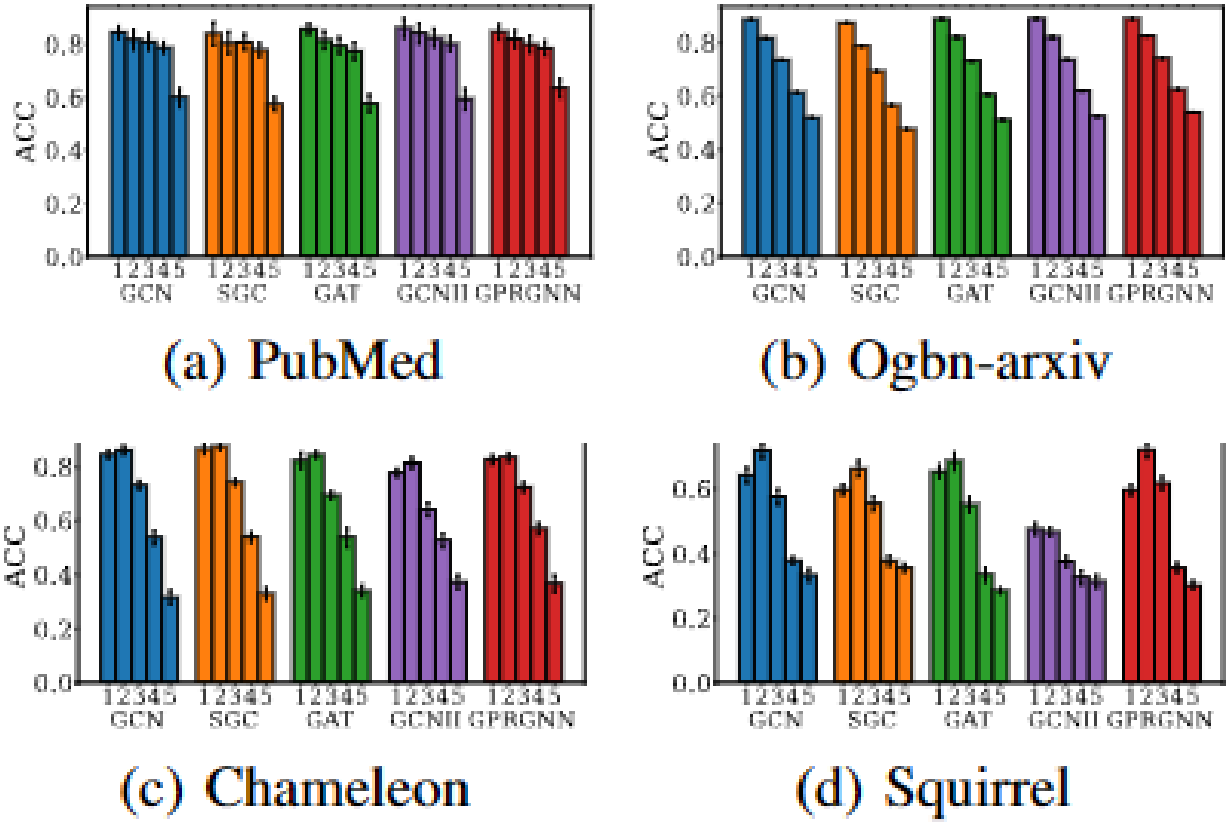


(a) PubMed (b) Ogbn-arxiv

(c) Chameleon (d) Squirrel

Additional higher-order performance disparity

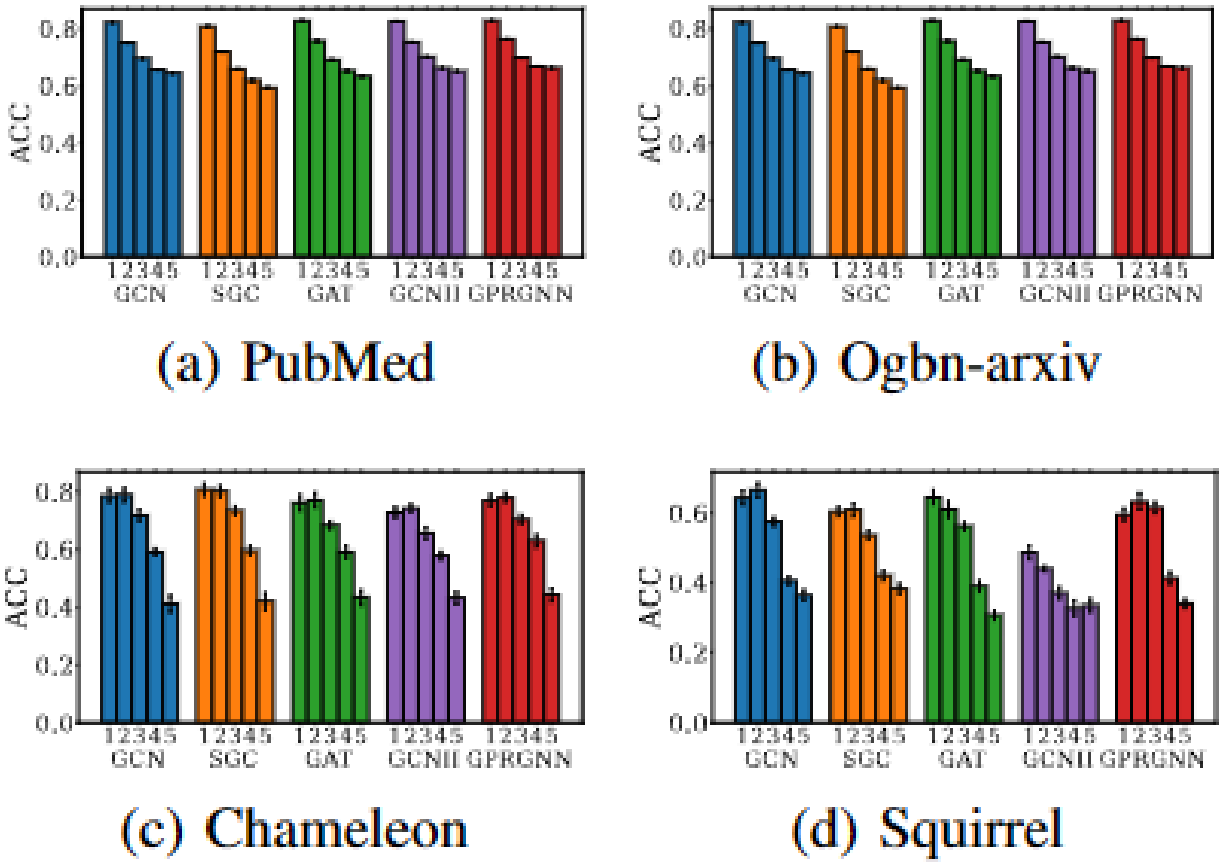
$$s = \epsilon_m$$



(a) PubMed (b) Ogbn-arxiv (c) Chameleon (d) Squirrel

Additional higher-order performance disparity

$$s = |h_{\text{tr}} - h_m|$$



(a) PubMed (b) Ogbn-arxiv

(c) Chameleon (d) Squirrel

OOD statistics

Table 14: the numbers of train, validation, test nodes on OOD data split

Dataset	Cora	CiteSeer	PubMed	Arxiv	Squirrel	Chameleon
#train	1599	1160	12466	85788	3709	1642
#valid	400	290	3117	21447	928	441
#test	486	660	4134	62108	564	564

No observe covariance shift

Table 15: MMD distance between train and validation, test sets on both i.i.d. and ood settings.

Dataset	Cora	CiteSeer	PubMed	Arxiv	Chameleon	Squirrel
IID valid	0.565	0.345	0.082	0.149	0.951	1.04
IID test	0.610	0.600	0.050	0.276	0.882	0.92
OOD valid	0.564	0.233	0.127	0.211	0.977	1.192
OOD test	0.597	0.598	0.442	0.420	0.854	0.92

Table 17: Train and test homophily ratios on the OOD datasets in [61]

	Twitch-explicit	FaceBook-100	Ogb-arxiv	elliptic	Cora	Amazon-photo
Train Homo	0.53	0.18	0.38	0.12	0.69	0.90
Test Homo	0.53	0.54	0.42	0.57	0.69	0.90