Dissecting the Failure of Invariant Learning on Graphs

Qixun Wang[1]   **Yifei Wang [2]**   **Yisen Wang [1,3]**∗   **Xianghua Ying [1]**
[1] State Key Laboratory of General Artificial Intelligence,
School of Intelligence Science and Technology, Peking University
[2] CSAIL, MIT
[3] Institute for Artificial Intelligence, Peking University

Abstract

Enhancing node-level Out-Of-Distribution (OOD) generalization on graphs remains a crucial area of research. In this paper, we develop a Structural Causal Model (SCM) to theoretically dissect the performance of two prominent invariant learning methods—Invariant Risk Minimization (IRM) and Variance-Risk Extrapolation (VREx)—in node-level OOD settings. Our analysis reveals a critical limitation: due to the lack of class-conditional invariance constraints, these methods may struggle to accurately identify the structure of the predictive invariant ego-graph and consequently rely on spurious features. To address this, we propose Cross-environment Intra-class Alignment (CIA), which explicitly eliminates spurious features by aligning cross-environment representations conditioned on the same class, bypassing the need for explicit knowledge of the causal pattern structure. To adapt CIA to node-level OOD scenarios where environment labels are hard to obtain, we further propose CIA-LRA (Localized Reweighting Alignment) that leverages the distribution of neighboring labels to selectively align node representations, effectively distinguishing and preserving invariant features while removing spurious ones, all without relying on environment labels. We theoretically prove CIA-LRA's effectiveness by deriving an OOD generalization error bound based on PAC-Bayesian analysis. Experiments on graph OOD benchmarks validate the superiority of CIA and CIA-LRA, marking a significant advancement in node-level OOD generalization. The codes are available at `https://github.com/NOVAglow646/NeurIPS24-Invariant-Learning-on-Graphs`.

1 Introduction

Generalizing to unseen testing distributions that differ from the training distributions, known as Out-Of-Distribution (OOD) generalization, is one of the key challenges in machine learning. Invariant learning, which aims to capture predictive features that remain consistent under distributional shifts, is a crucial strategy for addressing OOD generalization. Numerous invariant learning methods have been proposed to tackle OOD problems in computer vision (CV) tasks [Arjovsky et al., 2020, Krueger et al., 2021, Bui et al., 2021, Rame et al., 2022, Wang et al., 2019, Mahajan et al., 2021, Zhang et al., 2021a, Wang et al., 2022a, Yi et al., 2022, Wang et al., 2022b, Xin et al., 2023]. While in recent years, enhancing OOD generalization on graph data is an emerging research direction attracting increasing attention. In this work, we focus on the challenge of node-level OOD generalization on graphs.

Straightforwardly adapting the above methods to node-level graph OOD scenarios presents several challenges: 1) the prediction of a node's label depends on its neighbored samples in an ego-subgraph, causing a gap from conventional CV OOD scenarios where samples are independently predicted; and 2) environment labels in node-level tasks are often unavailable [Wu et al., 2021, Li et al., 2023a,

---

∗Corresponding author: Yisen Wang (yisen.wang@pku.edu.cn).

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

Liu et al., 2023], rendering invariant learning methods based on environment partitioning infeasible. To illustrate the failure of directly adapting traditional invariant learning to graphs, we evaluate two representative OOD approaches, Invariant Risk Minimization (IRM [Arjovsky et al., 2020]) and Variance-Risk Extrapolation (VREx [Krueger et al., 2021]), in OOD node classification scenarios. We choose IRM and VREx for two reasons: 1) Numerous node-level graph OOD strategies [Zhang et al., 2021b, Wu et al., 2021, Liu et al., 2023, Tian et al., 2024] utilize VREx as invariance regularization (details in Appendix A.2). Therefore, analyzing VREx can cover a significant portion of graph-OOD methods; and 2) IRM and VREx are two prominent OOD methods that we can theoretically prove to be effective on non-graph data (Proposition 2.2). By testing their performance on graph data, we can better reveal the differences between graph and non-graph data. We choose real-world graph datasets: Arxiv, Cora, and WebKB; synthetic datasets: CBAS and a toy graph OOD dataset with spurious correlations between node features and labels for evaluation. From Table 1, we observe that IRM and VREx offer marginal or no improvement over ERM on both real-world and synthetic node-level graph OOD datasets. This naturally raises some questions here:

> *On graphs, why do traditional invariant learning methods fail? How to make them work again?*

To theoretically analyze their failure modes, we build a Structural Causal Model (SCM) to model the data generation process under two types of distributional shifts: concept shift and covariate shift, and gain a high-level understanding of the challenges in node-level OOD generalization: To correctly predict a node's label, the structure of a predictive invariant neighboring ego-graph (which we call it a *causal pattern*) and their invariant node features must be learned. However, identifying the correct structure of the causal pattern presents additional optimization requirements (compared to CV scenarios) for Graph Neural Networks (GNNs) since they must adjust the aggregation parameters (such as the attention weights in GAT [Veličković et al., 2018]) to achieve this. IRM and VREx lack class-conditional invariance constraints, which causes insufficient supervision for regularizing the training of these aggregation parameters, leading to non-unique solutions of GNN parameters and potentially resulting in the learning of spurious features. (detailed analysis is in Section 2). To overcome this, we propose Cross-environment Intra-class Alignment (**CIA**) that further considers class-conditional invariance to identify causal patterns[2]. We theoretically demonstrate that by aligning node representations of the same class and different environments, CIA can eliminate spurious features and learn the correct causal pattern, as same-class different-environment samples share similar causal patterns while exhibiting different spurious features. Table 1 shows CIA's empirical gains. To leverage the advantage of CIA and adapt it to scenarios without environment labels, we further propose **CIA-LRA** (Localized Reweighting Alignment), utilizing localized label distribution to find node pairs with significant differences in spurious features and small differences in causal ones for alignment, to eliminate the spurious features while alleviating the collapse of the causal ones. Our contributions are summarized as follows:

Algorithms	Large-Cov.	Large-Con.	Toy
ERM	57.74	59.57	33.6
IRM	57.59	59.46	34.9
VREx	58.46	59.83	33.9
CIA (ours)	59.68	60.89	37.0
CIA-LRA (ours)	**61.94**	**63.03**	**39.1**

Table 1: *Real-Cov./Con.* are average OOD accuracy on the covariate/concept shift of Arxiv, Cora, CBAS, and WebKB. *Toy* denotes results on our toy synthetic graph OOD dataset.

1. By constructing an SCM, we provide a theoretical analysis revealing that VREx and IRM could rely on spurious features when using a GAT-like GNN (Section 2.2) in node-level OOD scenarios, revealing a key challenge of invariant learning on graphs.

2. We propose CIA and theoretically prove its effectiveness in learning invariant representations on graphs (Section 3.1). To adapt CIA to node-level OOD scenarios where environment labels are unavailable, we further propose CIA-LRA that requires no environment labels or complex environmental partitioning processes to achieve invariant learning (Section 3.2), with theoretical guarantees on its generalization performance (Section 4).

3. We evaluate CIA and CIA-LRA on the Graph OOD benchmark (GOOD) [Gui et al., 2022] on GAT and GCN [Kipf and Welling, 2016]. The results demonstrate CIA's superiority over

---

[2]Although numerous node-level OOD strategies from different perspectives have been proposed such as GNN architecture design [Wu et al., 2024] or feature augmentation [Li et al., 2023b], we focus on developing an invariant learning objective that could be integrated into and improve other graph-OOD frameworks in a plug-and-play manner (validated in Section 5.3), serving as a general solution.

2

non-graph invariant learning methods, and CIA-LRA achieves state-of-the-art performance (Section 5.2).

We leave comparisons of our method and existing node-level OOD works in Appendix A.3.

2 Dissecting Invariant Learning on Graphs

For OOD node classification, we are given a single training graph $\mathcal{G} = (A, X, Y)$ containing N nodes $\mathcal{V} = \{v_i\}_{i=1}^N$ from multiple training environments $e \in \mathcal{E}_{\text{tr}}$. $A \in \{0, 1\}^{N \times N}$ is the adjacency matrix, $A_{i,j} = 1$ iff there is an edge between v_i and v_j. $X \in \mathbb{R}^{N \times D}$ are node features. The i-th row $X_i \in \mathbb{R}^D$ represents the original node feature of v_i. $Y \in \{0, 1, ..., C-1\}^N$ are the labels, C is the number of the classes. Denote the subgraph containing nodes of environment e as $\mathcal{G}^e = (A^e, X^e, Y^e)$, which follows the distribution p_e. Let $A^e \in \{0, 1\}^{N^e \times N^e}$ and D^e be the adjacency matrix and the diagonal degree matrix for nodes from environment e respectively, where $D_{ii}^e = \sum_{j=1} A^e_{ij}$, N^e is the number of samples in e. Denote the normalized adjacency matrix as $\bar{A}^e = (D^e + I_{N^e})^{-\frac{1}{2}} A^e (D^e + I_{N^e})^{-\frac{1}{2}}$, I_{N^e} is the identity matrix. Let $\tilde{A}^e = \bar{A} + (D^e + I_{N^e})^{-\frac{1}{2}} I_{N^e} (D^e + I_{N^e})^{-\frac{1}{2}}$. Suppose the unseen test environments are $e' \in \mathcal{E}_{\text{te}}$. The test distribution $p_{e'} \neq p_e \forall e' \in \mathcal{E}_{\text{te}}$, $\forall e \in \mathcal{E}_{\text{tr}}$. OOD generalization aims to minimize the prediction error over test distributions.

To understand the obstacles in invariant learning on graphs, we start by examining whether IRMv1 (practical implementation of the original challenging IRM objective, proposed by Arjovsky et al. [2020]) and VREx can be successfully transferred to node-level graph OOD tasks. Their objectives are as follows:

$$\begin{aligned} \text{(IRMv1)} \ & \min_{w,\phi} \mathbb{E}_e[\mathcal{L}\left(w \circ \phi(X^e), Y^e\right) + \beta \|\nabla_{w|w=1.0}\mathcal{L}\left(w \circ \phi(X^e), Y^e\right)\|_2^2], \\ \text{(VREx)} \ & \min_{w,\phi} \mathbb{E}_e[\mathcal{L}\left(w \circ \phi(X^e), Y^e\right)] + \beta \text{Var}_e[\mathcal{L}\left(w \circ \phi(X^e), Y^e\right)], \end{aligned} \tag{1}$$

where w and ϕ denote the classifier and feature extractor, respectively. $\mathcal{L}$ is the cross-entropy loss. β is some hyperparameter.

2.1 A Causal Data Model on Graphs

Data generation process. We construct an SCM to characterize two kinds of distribution shifts: concept shift (Figure 1a) and covariate shift (Figure 1b). C and S denote unobservable causal/spurious latent variables that affect the generation of the graph G, and dashed E are environmental variables usually unobservable. We consider a simple case that each node v in environment e has a 2-dim feature $[x_v^1, x_v^2]^\top$, $x_v^1, x_v^2 \in \mathbb{R}$. Denote the concatenated node features of all nodes in e as $X_1 \in \mathbb{R}^{N^e \times 1}$ and $X_2^e \in \mathbb{R}^{N^e \times 1}$ corresponding to x_v^1 and x_v^2, respectively. For the SCM in Figure 1a[3], the data generation process of environment e is

(a) concept shift (b) covariate shift

Figure 1: Causal graphs of the SCMs considered in our work.

$$Y^e = (\tilde{A}^e)^k X_1 + n_1, \ X_2^e = (\tilde{A}^e)^m Y^e + n_2 + \epsilon^e, \tag{2}$$

where $k \in \mathbb{N}^+$ represents the "depth" (number of hops) of the causal pattern, and $m \in \mathbb{N}^+$ is the depth of the ego-graph determining the spurious node features. $n_1 \in \mathbb{R}^{N^e \times 1}$ and $n_2 \in \mathbb{R}^{N^e \times 1}$ represent random Gaussian noise. ϵ^e stands for an environmental variable, causing the spurious correlations between X_2^e and Y. A detailed description of the model is in Appendix F.1.

How the proposed model considers both node feature shifts and structural shifts? X_1 represent invariant node features causing Y^e. X_2^e denotes spurious node features that vary with environments. As for structural shifts, we consider an environmental $\tilde{A}^e$ in Equation (2), which means the structure can vary with environments. For example, there could be a spurious correlation between certain structures and the label; or, the graph connectivity or size may shift [Buffelli et al., 2022, Xia et al.,

---

[3]Due to space limitation, we only present the concept shift case in the main text and leave the covariate shift case in Appendix B. Our following results in Section 2 hold for both shifts.

3

2023]. We model the invariant structural feature as the structure of a node's k-layer neighboring ego-graph. See Appendix F.2 for more discussions of the structural shifts.

We also have the following assumption about the stability of the causal patterns across environments:

Assumption 2.1. (Stability of the causal patterns) The k-layer causal pattern in Equation (2) is invariant across environments for every class c.

A simple multi-layer GNN. Consider a L-layer GNN f parameterized by $\Theta = \left\{ \theta_1, \theta_2, \theta_1^{1\,(l)}, \theta_1^{2\,(l)}, \theta_2^{1\,(l)}, \theta_2^{2\,(l)} \right\}, l = 1, 2, ..., L-1$:

$$f_\Theta(A, X) = H_1^{(L)}\theta_1 + H_2^{(L)}\theta_2, \text{ where}$$

$$\begin{pmatrix} H_1^{(l)} & H_2^{(l)} \end{pmatrix} = \begin{pmatrix} \bar{A} & \bar{I} \end{pmatrix} \begin{pmatrix} \theta_1^{1\,(l-1)} & \theta_2^{1\,(l-1)} \\ \theta_1^{2\,(l-1)} & \theta_2^{2\,(l-1)} \end{pmatrix} \begin{pmatrix} H_1^{(l-1)} & 0 \\ 0 & H_2^{(l-1)} \end{pmatrix}, \ l = 2, ..., L, \ H_1^{(1)} = X_1, \ H_2^{(1)} = X_2,$$
(3)

where θ_i and $\theta_j^{i\,(l)}$ are scalars for $i, j \in \{1, 2\}, \forall l$. $H_i^{(l)} \in \mathbb{R}^{N \times D}$ are GNN representations.

Remark. In this GNN, we keep only the top-layer weight matrix $[\theta_1\ \theta_2]^\top$, and let the weight matrix of lower layers $1, ..., L-1$ be an identity matrix. This architecture resembles an SGC [**?**]. θ_1 and θ_2 are for invariant/spurious features, respectively. $\theta_1^{1\,(l)}, \theta_2^{1\,(l)}$ are weights for aggregating features from neighboring nodes and $\theta_1^{2\,(l)}, \theta_2^{2\,(l)}$ are weights for features of a centered node, this setup can be seen as a GAT. When all lower-layer parameters equal 1, the GNN degenerates to a GCN (see Appendix F.2 for justification of the choice of the GNN).

We consider a regression problem that we aim to minimize the MSE loss over all environments $\mathbb{E}_e[R(e)] = \mathbb{E}_e \left[\mathbb{E}_{n_1, n_2} \left[\|f_\Theta(A^e, X^e) - Y^e\|_2^2 \right] \right]$. The optimal invariant parameter set Θ^* is

$$\begin{cases} \theta_1 = 1 \\ \theta_2 = 0 \quad \text{or} \quad \exists l \in \{1, ..., L-1\} \text{ s.t. } \theta_2^{1\,(l)} = \theta_2^{2\,(l)} = 0 \\ \theta_1^{1\,(l)} = 1, \theta_1^{2\,(l)} = 1, \quad l = L-1, ..., L-k+1 \\ \theta_1^{1\,(l)} = 0, \theta_1^{2\,(l)} = 1, \quad l = L-k, L-k-1, ..., 1 \end{cases}.$$
(4)

In Equation (4), the GNN parameters for spurious features (line 2) is zero, which means it removes spurious node features. Also, it learns the correct depth k of the causal pattern $\tilde{A}^k X_1$ (line 3-4).

2.2 Intriguing Failure of VREx and IRM on Graphs

Now we are ready to present the failure cases in this node-level OOD task: optimizing IRMv1 and VREx induces a model that relies on spurious features X_2^e to predict, leading to poor OOD generalization. To illustrate that this failure arises from the graph data, we first prove that IRMv1 and VREx can learn invariant features under the non-graph version of SCM of Equation (2).

Proposition 2.2. (IRMv1 and VREx can learn invariant features for non-graph tasks, proof is in Appendix G.1.1.) *For the non-graph version of the SCM in Equation (2),*

$$Y^e = X_1 + n_1, \ X_2^e = Y^e + n_2 + \epsilon^e,$$
(5)

VREx and IRMv1 can learn invariant features when using a linear network: $f(X) = \theta_1 X_1 + \theta_2 X_2$.

Now we will give the main theorem revealing the failure of VREx and IRMv1 on graphs.

Theorem 2.3. (IRMv1 and VREx will use spurious features on graphs, informal) *Under the SCM of Equation (2), the IRMv1 and VREx objectives have non-unique solutions for parameters of the GNN (3), and there exist solutions that use spurious features, i.e. $\theta_2 \neq 0$.*

Intuitive illustration of the failure. From Theorem 2.3, we find that the main reason for the failure lies in the message-passing mechanism in representation learning. Let's provide some key steps in the proof of the IRMv1 case as an illustration. For the non-graph OOD task Equation (5), we can verify that when IRMv1 objective is solved, i.e. $\nabla_w R(e) = 0$ for all e, the invariant solution $\theta_2 = 0$

4

leads to $(\theta_1)^2 X_1^\top X_1 - \theta_1 X_1^\top X_1 = 0$, which can be satisfied when $\theta = 1$. However, in the graph case, $\theta_2 = 0$ leads to

$$(\theta_1)^2((\tilde{A}^e)^s X_1)^\top (\tilde{A}^e)^s X_1 - \theta_1((\tilde{A}^e)^k X_1)^\top (\tilde{A}^e)^s X_1 = 0, \ \forall e, \tag{6}$$

where $(\tilde{A}^e)^s X_1$ is the learned representation of the GNN, $0 < s \leq L$. k is the depth of the causal pattern. Now we explain why the invariant solution may not hold on graphs. When the depth of the learned aggregation pattern $s \neq k$, Equation (6) cannot hold for a fixed θ_1 (since θ_1 will depend on e then). This means that identifying the underlying structure of the causal pattern imposes additional difficulty for invariant learning. Moreover, even if the GNN can learn representations of different depths (e.g. GAT)[4], the proof in Appendix G.1.3 shows that IRM failed to provide sufficient supervision to optimize the aggregation parameters θ^i_j, $i, j \in \{1, 2\}$ such that $s = k$. A similar analysis holds for VREx. In general, successful invariant learning on graphs requires capturing both invariant node features and the structure of the causal pattern, while methods like IRM and VREx that solely enforce a cross-environment invariance at the loss level[5] may not be able to achieve these goals. The formal versions and proof are in Appendix G.1.3 (IRM) and G.1.2 (VREx).

3 The Proposed Methods

3.1 Cross-environment Intra-class Alignment

Inspired by the examples of VREx and IRMv1, we aim to introduce additional invariance regularization to guide the model in identifying the underlying invariant node features and structures. We propose CIA (Cross-environment Intra-class Alignment), which aligns the representations from the same class across different environments. Intuitively, since such node pairs share similar invariant features and causal pattern structures while differ in spurious features, aligning their representations will help achieve our targets. Denote the representation of node i as $\phi_\Theta(i)$ and the classifier parameterized by θ_h as h_{θ_h} CIA's objective is:

$$\min_{\theta_h, \Theta} \mathbb{E}_e \left[\mathcal{L}(h_{\theta_h} \circ \phi_\Theta(A^e, X^e), Y^e) \right] \quad \text{s.t.} \quad \min_\Theta \mathcal{L}_{\text{CIA}} = \mathbb{E}_{\substack{e, e' \\ e \neq e'}} \mathbb{E}_c \mathbb{E}_{\substack{i, j \\ (i,j) \in \Omega^{e,e'}_c}} \left[\mathcal{D}(\phi_\Theta(i), \phi_\Theta(j)) \right] \tag{7}$$

where $\Omega^{e,e'}_c = \{(i,j) | i \neq j \wedge Y^e_i = Y^{e'}_j = c \wedge E_i = e, \ E_j = e'\}$ is the set of nodes with same label c and from two different environments. $\mathcal{L}$ is the cross-entropy loss. $\mathcal{D}$ is some distance metric and we adopt L-2 distance. Now we prove that CIA can learn invariant representations regardless of the unknown causal patterns:

Theorem 3.1. *Under the SCM of Equation (2) and Assumption 2.1, optimizing the CIA objective will lead to the optimal invariant solution Θ^* in Equation (4) for parameters of the GNN (3).*

The proof is in Appendix G.1.4. By enforcing class-conditional invariance, which is not considered in VREx and IRMv1, CIA overcomes the above obstacles and eliminates spurious features. As long as a GNN has the capacity to adaptively learn the true depth of the causal pattern (such as the one considered in Equation (3)) or a GAT, CIA can identify the invariant causal pattern.

Remark. One might note that the objective of CIA is analogous to MatchDG [Mahajan et al., 2021]. However, we are the first to adapt such an idea to node-level OOD tasks and theoretically reveal its advantage. In Appendix A.1, we compare our extension and the original MatchDG in detail.

3.2 Localized Reweighting Alignment: an Adaptation to Graphs without Environment Labels

So far, we have theoretically and empirically validated CIA's advantage on graphs, but it still requires environmental labels that are challenging to obtain in most node classification tasks [Wu et al., 2021, Liu et al., 2023, Li et al., 2023a]. In this section, we propose CIA-LRA (Localized Reweighting Alignment) that realizes CIA's objective without using environment labels by identifying node pairs with significant/minor differences in spurious/invariant features and then aligning their representations. As illustrated in Figure 2, CIA-LRA mainly incorporates three components:

---

[4]If the GNN is a GCN that has a fixed aggregation depth L, i.e. $s = L$, it will be even impossible to learn the true causal pattern if we choose an $L \neq k$ in advance.

[5]IRM minimizes the loss gradient w.r.t. the classifier in each environment, and VREx minimizes the loss variance across environments.

5

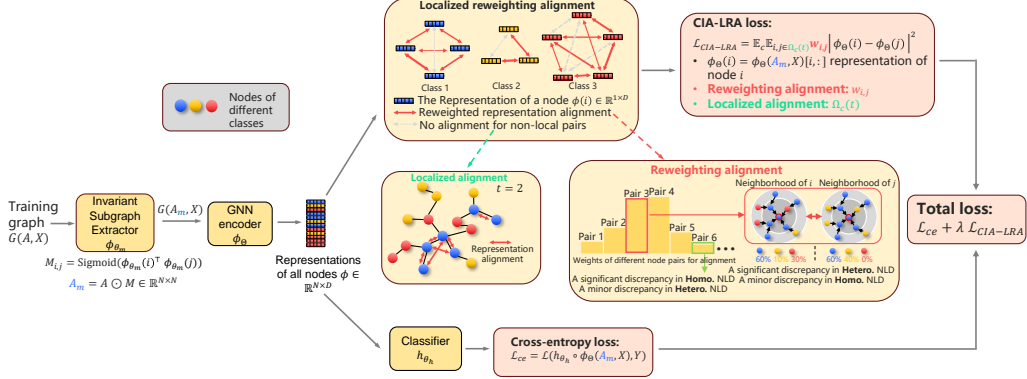



Figure 2: The overall framework of our proposed CIA-LRA. The invariant subgraph extractor ϕ_{θ_m} identifies the invariant subgraph for each node. Then the GNN encoder ϕ_Θ aggregates information from the estimated invariant subgraphs to output node representations. CIA-LRA mainly contains two strategies: localized alignment and reweighting alignment. Localized alignment: we restrict the alignment to a local range to avoid overalignment that may cause the collapse of invariant features (shown in Appendix D.1). Reweighting alignment: to better eliminate spurious features and preserve invariant features without using environment labels, we assign large weights to node pairs with significant discrepancies in heterophilic Neighborhood Label Distribution (NLD) and minor discrepancies in homophilic NLD. See Section 3.2 for a detailed analysis of CIA-LRA.

Localized alignment. To avoid learning a collapsed representation of invariant features, it is crucial to align node pairs that share similar invariant features. To achieve this, we align nodes close to each other (about 2 to 6 hops). This is based on two observations. First, we observe that spurious features tend to exhibit larger changes within local graph areas than invariant ones, and nodes from the same class that are too distant may differ more in their invariant features than the closer ones (evidence in Appendix D.4). This is because invariant features are generally more stable than spurious ones, according to Chen et al. [2022], Schölkopf et al. [2021]. Second, we empirically find that alignment over an extensive range or too many nodes yields only marginal performance improvements, or even leads to performance degradation (see Appendix D.1), while increasing computational costs. This may also be attributed to the feature collapse caused by excessive alignment of too many node pairs. Formally, the local pairs are defined as $\Omega_c(t) = \{(i,j)|i \neq j \wedge Y_i = Y_j = c \wedge d(i,j) \leq t\}$, where $d(i,j)$ represents the the shortest path length from node v_i to v_j, $t \in \mathbb{N}^+$ is a hyperparameter. Also, we propose to assign smaller weights to pairs more distant away from each other.

Reweighting Alignment. Since environment labels are unavailable, we need a metric to reflect the distribution of the spurious and invariant representations so that node pairs with significant/small differences in spurious/invariant features can be identified. Since we assume the causal patterns of the same class are similar (Assumption 2.1), the label distribution of homophilic (i.e., same-class) neighbors directly affects the invariant features aggregated to the centered node (empirical evidence in Appendix D.5.2). Therefore, pairs with smaller differences in the ratio of homophilic neighbors should be assigned larger weights for alignment. The ratio discrepancy can be calculated as follows: $r^{\text{same}}(c)_{i,j} = \left|r_i^c - r_j^c\right|$, where r_i^c is the ratio of the neighbors of v_i of class c within L hops (L is the number of layers of the GNN). As for spurious features, we utilize *Heterophilic* (i.e., different-class) *Neighborhood Label Distribution* (HeteNLD) as a measurement, as it affects the two kinds of main distributional shifts on graphs: 1) environmental node feature shifts, and 2) *Neighborhood Label Distribution* (NLD) shift (both empirically verified in Appendix D.5.1). HeteNLD determines the first kind of shift when correlations exist between labels and spurious node features, e.g., concept shift. The second kind, NLD shift, which is affected by HeteNLD, can be regarded as a structural shift as the discrepancy in neighborhood distribution will induce a gap in the aggregated representations (Theorem 4.4 shows this shift increases OOD error). Although aligning the representations significantly differing in homophilic neighbor ratio mitigates these two kinds of shifts, it also leads to the collapsed invariant representations and suboptimal performance (Table 4 shows this effect). Therefore we assign larger weights to the pair with a larger discrepancy in HeteNLD when alignment. The discrepancy in HeteNLD is calculated as follows: $r^{\text{diff}}(c)_{i,j} = \sum_{c' \neq c} \left|r_i^{c'} - r_j^{c'}\right|$.

6

Invariant Subgraph Extractor. In Section 2.1, the structural invariant features are defined as the k-hop neighboring ego-graph for ease of analysis. However, in practice, the invariant structure may merely be a subgraph of the neighborhood nodes. To better capture the invariant ego subgraph, we train an invariant subgraph extractor inspired by Li et al. [2023a]. Concretely, we learn an auxiliary GNN encoder ϕ_{θ_m} (parameterized by θ_m) to predict an soft edge mask $M \in \mathbb{R}^{N \times N}$, and then apply it during training and test:

$$M_{i,j} = \text{Sigmoid}(\phi_{\theta_m}(i)^\top \phi_{\theta_m}(j)), \ A_m = A \odot M \text{ , where } \odot \text{ is sample-wise multiplication.} \quad (8)$$

Now we are ready to present the formal objective of CIA-LRA:

$$\min_{\theta_h, \Theta, \theta_m} \mathcal{L}(h_{\theta_h} \circ \phi_\Theta(A_m, X), Y) \text{ s.t. } \min_{\Theta, \theta_m} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega_c(t)}} [w_{i,j} \mathcal{D}(\phi_\Theta(i), \phi_\Theta(j))],$$

$$\text{where } w_{i,j} = \text{Norm}\left(\frac{r^{\text{diff}}(c)_{i,j}}{d(i,j) r^{\text{same}}(c)_{i,j}}\right), \ \text{Norm}(\cdot) \text{ denotes a Min-Max normalization.} \quad (9)$$

In Equation (9), $\mathcal{L}$ is the cross-entropy loss, $\Omega_c(t)$, $r^{\text{diff}}(c)_{i,j}$, $r^{\text{same}}(c)_{i,j}$ and $d(i,j)$ are defined in the above analysis. In practice, we use CIA-LRA as a regularization term added to the cross-entropy loss with a weight λ as a hyperparameter. The detailed implementation of CIA-LRA is in Appendix E.

4 Theoretical Justification: an OOD Generalization Error Bound

Now will derive an OOD generalization error bound to show that optimizing CIA-LRA can minimize OOD error. To achieve this, we adopt a PAC-Bayesian framework following Ma et al. [2021] and establish a Contextual Stochastic Block Model (CSBM, [Deshpande et al., 2018]) for OOD multi-classification. The proposed CSBM-OOD is as follows (more discussions are in Appendix F.4):

Definition 4.1. (CSBM-OOD). For node i of class c from environment e, its node feature $x_i \in \mathbb{R}^D$ consists of two parts, $x_i = [x_{\text{inv}}^\top; x_{\text{sp}}^\top]^\top$, where $x_{\text{inv}} \in \mathbb{R}^{\frac{D}{2}}$ sampled from the Gaussian distribution $\mathcal{N}(\mu_c, \sigma^2 I)$ is the invariant feature and $x_{\text{sp}} \in \mathbb{R}^{\frac{D}{2}}$ sampled from the $\mathcal{N}(\mu_c^e, \sigma^2 I)$ is the spurious feature. [6] Suppose $\{\mu_c\}$ and $\{\mu_c^e\}$ for all c and e form sets of orthonormal basis. We use p_i^{hm} to denote the homophilic ratio of node i's one-hop neighbors and use $p_i^{\text{ht}}(c')$ to denote the heterophilic ratio of node i's one-hop neighbors of class c' ($c \neq c'$). We assume $\Pr(y_i = c)$ are the same for all classes c.

The GNN model used for deriving the error bound (following Ma et al. [2021]): The GNN model has a 1-layer mean aggregation g that outputs the aggregated feature $g_i \in \mathbb{R}^D$ for node i. The GNN classifier h on top of g is a ReLU-activated L-layer MLP with $W_1, ..., W_L$ as parameters for each layer. h is from a function family $\mathcal{H}$. The prediction for node i is $h_i \in \mathbb{R}^C$ with $h_i[c]$ representing the predicted logit for class c. Denote the largest width of all the hidden layers as b.

Notations. Denote nodes of environment e as V_e. We consider the error of generalizing from a mixed training environment e^{tr} to any test environment $e^{\text{te}} \in \mathcal{E}_{\text{te}}$, where $V_{e^{\text{tr}}} := \cup_{e \in \mathcal{E}_{\text{tr}}} V_e$ represents all training nodes. To guarantee the generalization, we need to characterize the distance between $V_{e^{\text{te}}}$ and $V_{e^{\text{tr}}}$: define $\epsilon_{e^{\text{te}}, e^{\text{tr}}} = \max_{j \in V_{e^{\text{te}}}} \min_{i \in V_{e^{\text{tr}}}} \|g_i - g_j\|_2$ as the aggregated feature distance between the training and test subgroup. Define the number of nodes in environment e as N_e. We consider the margin loss of environment e that is used by Ma et al. [2021], Mao et al. [2023]: $\widehat{\mathcal{L}}_e^\gamma(h) := \frac{1}{N_e} \sum_{v_i \in V_e} \mathbf{1}[h_i[y_i] \leq \gamma + \max_{c \neq y_i} h_i[c]]$.

Now we introduce some assumptions adapted from Ma et al. [2021] that are used in our proof.

Assumption 4.2. (Equal-sized and disjoint near sets, adapted from Assumption 2 of Ma et al. [2021]) For each node $v_i \in V_{e^{\text{tr}}}$, define $V_{e^{\text{te}}}^{(i)} := \{j \in V_{e^{\text{te}}} \mid \|g_i - g_j\|_2 \leq \epsilon_{e^{\text{tr}}, e^{\text{te}}}\}$. For any test environment e^{te}, assume $V_{e^{\text{te}}}^{(i)}$ of each $v_i \in V_{e^{\text{tr}}}$ are disjoint and have the same size $N_{e^{\text{te}}} \in \mathbb{N}^+$.

Assumption 4.3. (concentrated expected loss difference, adapted from Assumption 3 of Ma et al. [2021]) Let P be a distribution on $\mathcal{H}$, defined by sampling the vectorized MLP parameters from

---

[6] For OOD scenarios that spurious features are not correlated with Y, we just need to let μ_c^e are the same for all c in the environment e, so this this definition is without loss of generality.

7

$\mathcal{N}\left(0, \sigma^2 I\right)$ for some $\sigma^2 \leq \frac{\left(\gamma/8\epsilon_{e^{\text{te}}, e^{\text{tr}}}\right)^{2/L}}{2b\left(\lambda N_{e^{\text{tr}}}^{-\alpha} + \ln 2bL\right)}$. For any L layer GNN classifier $h \in \mathcal{H}$ with model parameters $W_1^h, \ldots, W_L^h$, define $T_h := \max_{l=1,\ldots,L} \|W_l\|_2$. Assume that there exists some $0 < \alpha < \frac{1}{4}$ satisfying

$$\Pr_{h \sim P}\left(\mathcal{L}_{e^{\text{te}}}^{\gamma/4}(h) - \mathcal{L}_{e^{\text{tr}}}^{\gamma/2}(h) > N_{e^{\text{tr}}}^{-\alpha} + HC\epsilon_{e^{\text{te}}, e^{\text{tr}}} \,\middle|\, T_h^L \epsilon_{e^{\text{te}}, e^{\text{tr}}} > \frac{\gamma}{8}\right) \leq e^{-N_{e^{\text{tr}}}^{2\alpha}}$$

Now we present the node-level OOD generalization bound (proof in Appendix G.3):

Theorem 4.4. *(Subgroup OOD Generalization Bound for GNNs, informal). Let $\tilde{h}$ be any classifier in a function family $\mathcal{H}$ with parameters $\left\{\widetilde{W}_l\right\}_{l=1}^{L}$. Under Assumption 4.2 and 4.3, for any $e^{te} \in \mathcal{E}_{te}, \gamma \geq 0$, and large enough $N_{e^{tr}}$, there exist $0 < \alpha < \frac{1}{4}$ with probability at least $1 - \delta$, we have*

$$\mathcal{L}_{e^{te}}^0(\tilde{h}) \leq \widehat{\mathcal{L}}_{e^{tr}}^{\gamma}(\tilde{h}) + O\big(\underbrace{\frac{1}{\sigma^2}\big(\sum_{c=1}^{C}\sum_{c' \neq c}\big(\sqrt{|[(\mu_c - \mu_{c'})^{\top}; (\mu_c^{e^{te}} - \mu_{c'}^{e^{te}})^{\top}]|}\big)\big)\epsilon_{e^{te}, e^{tr}}}_{(a)}$$

$$+ \underbrace{2\sum_{c=1}^{C}(C-1)B_{e^{te}}|\mu_c^{e^{te}} - \mu_c^{e^{tr}}|)}_{(b)} + \underbrace{\frac{1}{2\sigma^2}\frac{1}{N_{e^{tr}}}\sum_{i \in V_{e^{tr}}}\frac{1}{N_{e^{te}}}\sum_{j \in V_{e^{te}}^{(i)}}\sum_{c=1}^{C}\sum_{c' \neq c}|p_j^{ht}(c'|c) - p_i^{ht}(c'|c)|)}_{(c)} + const$$

$$(10)$$

where $p_i^{ht}(c'|c)$ is the ratio of heterophilic neighbors of class c' when $y_i = c$, $B_{e^{te}} = \max_{i \in V_{e^{tr}} \cup V_{e^{te}}} \|g_i\|_2$ is the maximum feature norm, $V_{e^{te}}^{(i)} := \left\{j \in V_{e^{te}} \mid \|g_i - g_j\|_2 \leq \epsilon_{e^{tr}, e^{te}}\right\}$. const is a constant depending on α, δ, and γ.

The observations from Theorem 4.4 is summarized as follows: Term **(a)** reflects the separability of the original features of different classes $|[(\mu_c - \mu_{c'})^{\top}; (\mu_c^{e^{te}} - \mu_{c'}^{e^{te}})^{\top}]|$ and the distance of the aggregated features between the training and test set $\epsilon_{e^{te}, e^{tr}}$. The former factor is the nature of the dataset itself. Term **(b)** is the distributional discrepancy between the training and test subgroups, caused by the distribution shifts in spurious features. When there exist correlations between labels and spurious features, CIA-LRA can minimize this term by minimizing the representation distance of node pairs with large discrepancy in the label distribution of heterophilic neighbors[7]. Term **(c)** measures the shift in HeteNLD between the training and test subgroups, representing the OOD error caused by the shift in the aggregated features of the same class. CIA-LRA minimizes this term by enforcing stronger alignment on pairs with greater HeteNLD differences.

5 Experiments

5.1 Experiment Setup

We run experiments using 3-layer GAT and GCN on GOOD [Gui et al., 2022], a graph OOD benchmark. We reported the results on both covariate shift and concept shift. The detailed experimental setup and hyperparameter settings are in Appendix C. We compare our methods with the following algorithms: **ERM** [Vapnik, 1999]; traditional invariant learning methods: **IRM**, **VREx**, **GroupDRO** [Sagawa et al., 2019], **Deep Coral** [Sun and Saenko, 2016], **IGA** [Koyama and Yamaguchi, 2020]; graph OOD methods: **EERM**, **SRGNN**, **CIT** [Xia et al., 2023], **CaNet** [Wu et al., 2024]; graph data augmentation: **Mixup** [Wang et al., 2021a], **GTrans** [Jin et al., 2022].

5.2 Main Results of OOD Generalization

Table 2 reports the main OOD generalization results. The observations are summarized as follows: 1) CIA-LRA improves the best baseline methods by 2.44% and 3.23% on GAT and GCN, respectively,

---

[7]Although the term $|\mu_c^{e^{te}} - \mu_c^{e^{tr}}|$ is the cross-environment distance of the original data, it can be minimized implicitly by aligning the representation induced by the two subgroups. This is because minimizing the distance between learnable representations of two node groups is equivalent to using a fixed mean aggregation on two groups with closer original features (in Theorem 4.4 we fix the feature extractor to be a mean aggregation layer).

8

Table 2: OOD test accuracy (%). Our methods are marked in **bold**. The best and second-best results are shown in **bold** and underlined, respectively. The values in parentheses are standard deviations. 'OOM' denotes Out Of Memory. Results of EERM on Cora GCN marked by '*' are from Gui et al. [2022] since we got OOM during our running.

(a) Results on GAT

	Covariate shift							Concept shift						
	Arxiv		Cora		CBAS	WebKB		Arxiv		Cora		CBAS	WebKB	
Dataset							average							average
Domain	degree	time	degree	word	color	university		degree	time	degree	word	color	university	
							Non-graph-specific methods							
ERM	59.12(0.09)	71.53(0.16)	55.30(0.34)	63.75(0.39)	65.24(2.69)	31.48(6.12)	57.74	65.70(0.18)	65.79(0.05)	61.36(0.38)	64.38(0.13)	74.52(1.87)	25.69(1.30)	59.57
IRM	59.16(0.14)	71.60(0.13)	55.07(0.30)	63.75(0.26)	65.24(1.78)	30.69(8.63)	57.59	65.79(0.35)	66.24(0.20)	61.42(0.34)	64.31(0.39)	73.33(0.89)	25.69(1.98))	59.46
VREx	59.10(0.12)	71.41(0.20)	55.34(0.16)	64.13(0.05)	66.67(2.93)	34.13(8.27)	58.46	65.94(0.03)	66.16(0.22)	61.77(0.30)	64.02(0.22)	73.57(3.50)	27.52(6.74)	59.83
GroupDRO	59.04(0.15)	71.30(0.04)	55.03(0.45)	63.82(0.06)	67.62(2.43)	31.75(2.82)	58.09	65.93(0.22)	66.14(0.15)	61.31(0.20)	64.07(0.25)	73.81(1.78)	26.91(2.16)	59.70
Deep Coral	59.12(0.10)	71.43(0.12)	55.35(0.45)	63.96(0.07)	64.76(2.43)	33.86(6.93)	58.08	65.76(0.21)	66.25(0.19)	61.62(0.26)	64.26(0.28)	75.95(3.21)	30.27(3.26)	60.69
IGA	59.01(0.17)	71.49(0.32)	55.56(0.12)	65.07(0.25)	65.71(5.08)	29.89(6.56)	57.79	65.87(0.19)	65.93(0.16)	62.62(0.13)	64.56(0.23)	70.95(3.51)	28.14(2.16)	59.68
MatchDG	OOM	OOM	55.74(0.25)	65.30(0.11)	67.15(5.08)	35.45(8.10)	-	OOM	OOM	62.91(0.27)	64.83(0.08)	75.24(2.36)	29.36(5.40)	-
CIA	**59.26(0.04)**	**71.65(0.25)**	**56.34(0.35)**	**65.55(0.19)**	**69.05(3.75)**	**36.24(0.75)**	**59.68**	**65.96(0.07)**	**66.39(0.25)**	**62.72(0.22)**	**64.92(0.28)**	74.76(2.63)	**30.58(3.12)**	**60.89**
							Graph-specific methods							
EERM	OOM	OOM	46.63(1.75)	62.57(0.50)	60.47(4.10)	33.33(14.60)	-	OOM	OOM	48.05(2.03)	53.02(1.23)	60.95(3.56)	25.38(4.26)	-
SRGNN	58.76(0.20)	71.37(0.37)	55.87(0.32)	64.50(0.35)	68.09(0.67)	28.84(1.35)	57.91	65.87(0.35)	66.02(0.14)	61.21(0.29)	64.10(0.28)	72.38(1.22)	23.55(1.56)	58.86
Mixup	59.32(0.11)	71.78(0.08)	56.77(0.36)	65.70(0.28)	63.33(8.60)	20.37(11.38)	56.21	63.11(0.10)	65.33(0.30)	63.97(0.18)	65.42(0.32)	73.33(1.47)	38.53(0.75)	61.62
GTrans	OOM	OOM	51.49(0.23)	62.48(0.25)	61.90(3.56)	21.69(7.17)	-	OOM	OOM	60.93(0.17)	62.68(0.28)	73.57(2.10)	25.08(1.88)	-
CIT	OOM	OOM	53.13(2.05)	63.76(0.20)	61.43(3.09)	24.60(7.47)	-	OOM	OOM	60.89(0.36)	63.60(0.48)	70.24(1.68)	24.16(5.26)	-
CaNet	OOM	OOM	55.35(0.14)	62.76(0.25)	68.09(1.78)	23.81(15.16)	-	OOM	OOM	60.97(0.07)	63.73(0.44)	75.95(3.41)	24.77(3.97)	-
CIA-LRA	**59.44(0.10)**	**71.79(0.13)**	**57.95(0.13)**	**68.59(0.26)**	**75.24(1.78)**	**38.62(3.57)**	**61.94**	**66.41(0.22)**	**66.47(0.16)**	**67.08(0.26)**	**68.05(0.14)**	**78.34(3.51)**	**31.80(1.88)**	**63.03**

(b) Results on GCN

	Covariate shift							Concept shift						
	Arxiv		Cora		CBAS	WebKB		Arxiv		Cora		CBAS	WebKB	
Dataset							average							average
Domain	degree	time	degree	word	color	university		degree	time	degree	word	color	university	
							Non-graph-specific methods							
ERM	58.92(0.14)	70.98(0.20)	55.78(0.52)	64.76(0.30)	78.57(2.02)	16.14(1.35)	57.52	62.92(0.21)	67.36(0.07)	60.24(0.40)	64.32(0.15)	82.14(1.17)	27.52(0.75)	60.75
IRM	58.93(0.17)	70.86(0.12)	55.77(0.66)	64.81(0.33)	78.57(1.17)	13.75(4.91)	57.12	62.79(0.11)	67.42(0.08)	61.23(0.08)	64.32(0.22)	81.67(0.89)	27.52(0.75)	60.84
VREx	58.75(0.16)	69.80(0.21)	55.97(0.53)	64.43(0.38)	79.05(1.78)	17.72(11.27)	57.62	63.06(0.43)	67.42(0.07)	60.69(0.42)	64.32(0.22)	82.86(1.17)	27.52(1.50)	60.98
GroupDRO	58.87(0.00)	70.93(0.09)	55.64(0.50)	64.62(0.30)	79.52(0.67)	14.29(2.59)	57.31	62.98(0.53)	67.41(0.22)	60.59(0.36)	64.34(0.25)	82.38(0.67)	28.44(0.00)	61.02
Deep Coral	59.04(0.16)	71.04(0.07)	56.03(0.37)	64.75(0.26)	78.09(0.67)	11.90(1.72)	56.81	63.09(0.28)	67.43(0.24)	60.41(0.27)	64.34(0.17)	82.86(0.58)	26.61(0.75)	60.79
IGA	58.87(0.17)	70.99(0.33)	55.94(0.58)	64.89(0.38)	79.05(1.78)	15.87(2.82)	57.60	62.04(0.02)	66.07(0.19)	61.06(0.36)	64.32(0.15)	82.38(0.89)	28.44(1.50)	60.72
MatchDG	OOM	OOM	56.57(0.46)	64.72(0.45)	77.14(1.17)	16.14(5.88)	-	OOM	OOM	60.49(0.14)	64.71(0.33)	84.05(0.89)	27.83(2.41)	-
CIA	59.03(0.39)	71.10(0.15)	56.80(0.54)	65.07(0.52)	80.00(2.02)	18.25(2.33)	**58.38**	63.87(0.26)	67.62(0.04)	61.59(0.18)	64.61(0.11)	85.71(0.72)	28.75(0.87)	61.83
							Graph-specific methods							
EERM	OOM	OOM	56.88(0.32)*	61.98(0.10)*	40.48(9.78)	16.21(5.67)	-	OOM	OOM	58.38(0.04)*	63.09(0.36)*	61.43(1.17)	28.04(11.67)	-
SRGNN	58.47(0.00)	70.83(0.10)	57.13(0.25)	64.50(0.35)	73.81(4.71)	16.40(1.63)	56.86	62.80(0.25)	67.17(0.23)	61.21(0.29)	64.53(0.27)	80.95(0.67)	27.52(0.75)	60.70
Mixup	57.80(0.19)	71.62(0.11)	57.89(0.27)	65.07(0.22)	70.00(5.34)	16.67(1.12)	56.51	62.33(0.34)	65.28(0.43)	63.65(0.37)	64.45(0.12)	85.48(0.67)	30.28(1.50)	58.58
GTrans	OOM	OOM	52.70(0.52)	63.37(0.27)	72.38(2.43)	10.58(0.99)	-	OOM	OOM	59.74(0.14)	63.56(0.18)	78.81(1.47)	26.91(1.56)	-
CIT	OOM	OOM	56.14(0.45)	64.79(0.29)	75.24(2.43)	19.31(4.32)	-	OOM	OOM	60.12(0.30)	64.26(0.42)	83.10(0.89)	28.14(1.14)	-
CaNet	OOM	OOM	57.35(0.04)	64.66(0.36)	80.95(0.67)	15.61(5.51)	-	OOM	OOM	60.34(0.20)	64.65(0.39)	85.24(3.32)	26.30(0.43)	-
CIA-LRA	**59.85(0.14)**	**71.81(0.20)**	**58.40(0.59)**	**65.95(0.04)**	**82.86(1.17)**	**19.84(2.83)**	**59.79**	**64.34(0.65)**	67.52(0.10)	**63.71(0.32)**	**65.07(0.21)**	**94.53(0.33)**	**36.70(0.75)**	**65.31**

achieving state-of-the-art performance. 2) CIA outperforms IRM and VREx on all splits, which validates our theoretical findings in Section 2. Notably, it performs best among the non-graph-specific methods. 3) CIA-LRA improves CIA in most cases. This suggests that our reweighting strategy can enhance generalization on graphs even without environment labels. 4) MatchDG outperforms IRM and VREx on 12 out of 16 splits but underperforms CIA on average (averaged over 16 splits except on Arxiv, CIA: 57.56, MatchDG: 56.73).

5.3 CIA can be Integrated into and Improve other Graph-OOD Methods

We replace VREx with CIA in the loss function of EERM to show that CIA can improve generalization in a plug-and-play manner. Table 3 shows that this improves original EERM by a large margin or has comparable performances, indicating the performance of node-level OOD algorithms can be limited by VREx.

Algorithms	Cora degree	Cora word	CBAS	WebKB
EERM	47.34	57.80	60.71	29.36
EERM-CIA	57.27	62.37	65.01	29.50
CIA	**59.53**	**65.24**	**71.91**	**33.41**

Table 3: By replacing VREx in EERM with CIA (marked as *EERM-CIA*), the performance is significantly improved.

5.4 Empirical Understanding of the Role of CIA-LRA

A synthetic dataset. We construct a synthetic dataset (mentioned in Section 1) to validate the role of each module in CIA-LRA in eliminating spurious features and preventing the collapse of invariant representations. We generate a random graph and create a 4-class OOD classification task. Each node has a 4-dim feature, with the first/last two dimensions representing invariant/spurious features (details in Appendix C.3), so we can disentangle the learned invariant and spurious representations.

9

Figure 3 depicts the OOD accuracy, the variance of the invariant representation, and the norm of the spurious representation across training epochs for CIA and CIA-LRA. The observations are summarized below.

Algorithms	Acc. (%)
IRM	61.14
VREx	61.32
no $r^{\text{diff}}(c)_{i,j}$	63.22
no $\frac{1}{r^{\text{same}}(c)_{i,j}}$	63.91
no $\frac{1}{d(i,j)}$	63.64
no M	63.39
use $r^{\text{same}}(c)_{i,j}$ in numerator	62.70
full CIA-LRA	**65.42**

Table 4: Ablation study of CIA-LRA. Results are averaged on the four splits of Cora.

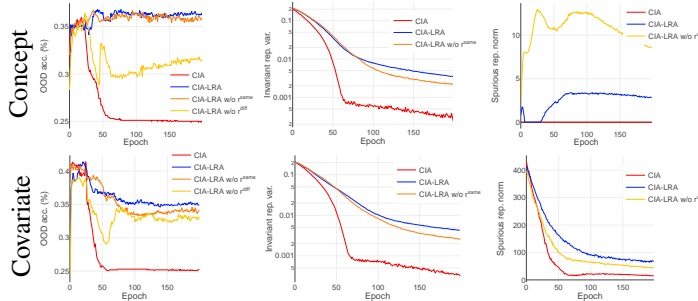



Figure 3: Left: OOD test accuracy. Mid: the variance of the invariant representation. Right: the norm of the spurious representation. CIA and CIA-LRA use $\lambda = 0.5$ in this figure.

1) Aligning the large discrepancy in HeteNLD helps to eliminate spurious features on concept shift and improves generalization. As evident from the right column, incorporating r^{diff} diminishes the norm of spurious features under concept shift. For covariate shift, while r^{diff} will not remove environmental spurious features due to their independence from labels, it still helps generalization since it reduces the error caused by shifts in HeteNLD as predicted by Theorem 4.4. 2) CIA-LRA alleviates collapse of causal representation that CIA may suffer when adopting a substantial λ. When using a large $\lambda (= 0.5)$, the performance of CIA deteriorates to the level of random guessing (25%) after approximately 50 epochs. In contrast, CIA-LRA sustains its accuracy at a high level because it avoids excessive alignment by aligning only local pairs and reweighting (further evidence in Appendix D.2). The mid column shows that the invariant features learned by CIA progressively collapse, even if CIA removes most spurious features (right column). 3) Maintaining the discrepancy in homophilic neighboring label distribution r^{same} helps keep the variance of the invariant representation, slightly improving performance.

Ablation study. We also conduct ablation studies on CIA-LRA. Table 4 shows that removing any module causes a significant performance drop, demonstrating the effectiveness of each module.

5.5 Effects of the Hyperparameters of CIA-LRA

This section analyzes the effect of λ and t of CIA-LRA. Figure 4 shows that the test accuracy increases with λ when $\lambda \leq 0.5$. Too small t leads to a sub-optimal performance due to insufficient regularization from aligning only a few pairs. Also, most parameter combinations outperform the baseline methods, indicating that CIA-LRA leads to consistently superior performance. Additional studies of the effects of λ and t are in Appendix D.2 and D.3, respectively.

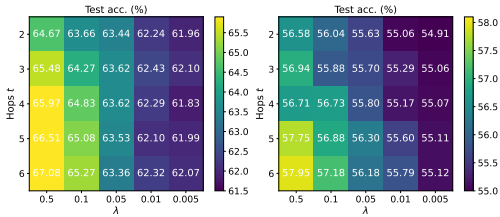
(a) Cora degree concept shift. ERM: 61.36, VREx: 61.42, IRM: 61.77.

(b) Cora degree covariate shift. ERM: 55.30, VREx: 55.07, IRM: 55.34.

Figure 4: Effect of λ and the number of hops t on OOD test accuracy (%).

6 Conclusion

In this work, by theoretically dissecting the failure of IRM and VREx in node-level graph OOD tasks, we attribute it to the difficulty in identifying the graph-specific causal pattern structures. To address this, we propose CIA with additional class-conditional invariance constraints and its environment-label-free variant CIA-LRA tailored for graph OOD scenarios. Further theoretical and experimental results validate their efficacy. Notably, CIA can be incorporated in other graph OOD frameworks, serving as a better invariant learning objective than the widely-used VREx on graphs.

10

Acknowledgement

Yisen Wang was supported by National Key R&D Program of China (2022ZD0160300), National Natural Science Foundation of China (92370129, 62376010), and Beijing Nova Program (20230484344, 20240484642). Xianghua Ying was supported by the National Natural Science Foundation of China (62371009).

References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. In *ICML*, 2020.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *ICML*, 2021.

Manh-Ha Bui, Toan Tran, Anh Tran, and Dinh Phung. Exploiting domain-specific features to enhance domain generalization. In *NeurIPS*, 2021.

Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In *ICML*, 2022.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *ICCV*, 2019.

Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In *ICML*, 2021.

Dinghuai Zhang, Kartik Ahuja, Yilun Xu, Yisen Wang, and Aaron Courville. Can subnetwork structure be the key to out-of-distribution generalization? In *ICML*, 2021a.

Ruoyu Wang, Mingyang Yi, Zhitang Chen, and Shengyu Zhu. Out-of-distribution generalization with causal invariant transformations. In *CVPR*, 2022a.

Mingyang Yi, Ruoyu Wang, Jiacheng Sun, Zhenguo Li, and Zhi-Ming Ma. Breaking correlation shift via conditional invariant regularizer. In *ICLR*, 2022.

Qixun Wang, Yifei Wang, Hong Zhu, and Yisen Wang. Improving out-of-distribution generalization by adversarial training with structured priors. In *NeurIPS*, 2022b.

Shiji Xin, Yifei Wang, Jingtong Su, and Yisen Wang. On the connection between invariant learning and adversarial training for out-of-distribution generalization. In *AAAI*, 2023.

Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. Handling distribution shifts on graphs: An invariance perspective. In *ICLR*, 2021.

Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. Invariant node representation learning under distribution shifts with multiple latent environments. *ACM Transactions on Information Systems*, (1):1–30, 2023a.

Yang Liu, Xiang Ao, Fuli Feng, Yunshan Ma, Kuan Li, Tat-Seng Chua, and Qing He. Flood: A flexible invariant learning framework for out-of-distribution generalization on graphs. In *SIGKDD*, 2023.

Shengyu Zhang, Kun Kuang, Jiezhong Qiu, Jin Yu, Zhou Zhao, Hongxia Yang, Zhongfei Zhang, and Fei Wu. Stable prediction on graphs with agnostic distribution shift. *arXiv preprint arXiv:2110.03865*, 2021b.

Qin Tian, Wenjun Wang, Chen Zhao, Minglai Shao, Wang Zhang, and Dong Li. Graphs generalization under distribution shifts. *arXiv preprint arXiv:2403.16334*, 2024.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018.

11

Qitian Wu, Fan Nie, Chenxiao Yang, Tianyi Bao, and Junchi Yan. Graph out-of-distribution generalization via causal intervention. *arXiv preprint arXiv:2402.11494*, 2024.

Xiner Li, Shurui Gui, Youzhi Luo, and Shuiwang Ji. Graph structure and feature extrapolation for out-of-distribution generalization. *arXiv preprint arXiv:2306.08076*, 2023b.

Shurui Gui, Xiner Li, Limei Wang, and Shuiwang Ji. Good: A graph out-of-distribution benchmark. In *NeurIPS*, 2022.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Davide Buffelli, Pietro Liò, and Fabio Vandin. Sizeshiftreg: a regularization method for improving size-generalization in graph neural networks. In *NeurIPS*, 2022.

Donglin Xia, Xiao Wang, Nian Liu, and Chuan Shi. Learning invariant representations of graph neural networks via cluster generalization. In *NeurIPS*, 2023.

Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. Learning causally invariant representations for out-of-distribution generalization on graphs. In *NeurIPS*, 2022.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, (5):612–634, 2021.

Jiaqi Ma, Junwei Deng, and Qiaozhu Mei. Subgroup generalization and fairness of graph neural networks. In *NeurIPS*, 2021.

Yash Deshpande, Subhabrata Sen, Andrea Montanari, and Elchanan Mossel. Contextual stochastic block models. In *NeurIPS*, 2018.

Haitao Mao, Zhikai Chen, Wei Jin, Haoyu Han, Yao Ma, Tong Zhao, Neil Shah, and Jiliang Tang. Demystifying structural disparity in graph neural networks: Can one size fit all? In *NeurIPS*, 2023.

Vladimir N Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, (5):988–999, 1999.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV Workshop*, 2016.

Masanori Koyama and Shoichiro Yamaguchi. When is invariance useful in an out-of-distribution generalization problem? *arXiv preprint arXiv:2008.01883*, 2020.

Yiwei Wang, Wei Wang, Yuxuan Liang, Yujun Cai, and Bryan Hooi. Mixup for node and graph classification. In *WWW*, 2021a.

Wei Jin, Tong Zhao, Jiayuan Ding, Yozen Liu, Jiliang Tang, and Neil Shah. Empowering graph representation learning with test-time graph transformation. *arXiv preprint arXiv:2210.03561*, 2022.

Yuge Shi, Jeffrey Seely, Philip HS Torr, N Siddharth, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. Gradient matching for domain generalization. *arXiv preprint arXiv:2104.09937*, 2021.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2020.

Junchi Yu, Jian Liang, and Ran He. Mind the label shift of augmentation-based graph ood generalization. In *CVPR*, 2023.

12

Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. Learning invariant graph representations for out-of-distribution generalization. In *NeurIPS*, 2022a.

Qi Zhu, Natalia Ponomareva, Jiawei Han, and Bryan Perozzi. Shift-robust gnns: Overcoming the limitations of localized graph training data. In *NeurIPS*, 2021.

Yongqiang Chen, Yatao Bian, Kaiwen Zhou, Binghui Xie, Bo Han, and James Cheng. Rethinking invariant graph representation learning without environment partitions. In *ICLR Workshop*, 2023a.

Fredrik D Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In *AISTATS*, 2019.

Ling Yang, Jiayi Zheng, Heyuan Wang, Zhongyi Liu, Zhilin Huang, Shenda Hong, Wentao Zhang, and Bin Cui. Individual and structural graph information bottlenecks for out-of-distribution generalization. *IEEE Transactions on Knowledge and Data Engineering*, 2023.

Ying-Xin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. Discovering invariant rationales for graph neural networks. *arXiv preprint arXiv:2201.12872*, 2022a.

Nianzu Yang, Kaipeng Zeng, Qitian Wu, Xiaosong Jia, and Junchi Yan. Learning substructure invariance for out-of-distribution molecular representations. In *NeurIPS*, 2022.

Tianrui Jia, Haoyang Li, Cheng Yang, Tao Tao, and Chuan Shi. Graph invariant learning with subgraph co-mixup for out-of-distribution generalization. *arXiv preprint arXiv:2312.10988*, 2023.

Haoyang Li, Xin Wang, Ziwei Zhang, and Wenwu Zhu. Ood-gnn: Out-of-distribution generalized graph neural network. *IEEE Transactions on Knowledge and Data Engineering*, 2022b.

Yongqiang Chen, Yatao Bian, Kaiwen Zhou, Binghui Xie, Bo Han, and James Cheng. Does invariant graph learning via environment augmentation learn invariance? In *NeurIPS*, 2023b.

Xuexin Chen, Ruichu Cai, Kaitao Zheng, Zhifan Jiang, Zhengting Huang, Zhifeng Hao, and Zijian Li. Unifying invariance and spuriousity for graph out-of-distribution via probability of necessity and sufficiency. *arXiv preprint arXiv:2402.09165*, 2024.

Shurui Gui, Meng Liu, Xiner Li, Youzhi Luo, and Shuiwang Ji. Joint learning of label and environment causal independence for graph out-of-distribution generalization. In *NeurIPS*, 2024.

David Burshtein, Vincent Della Pietra, Dimitri Kanevsky, and Arthur Nadas. Minimum impurity partitions. *The Annals of Statistics*, pages 1637–1646, 1992.

Bernhard Schölkopf. Causality for machine learning. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 765–804. 2022.

Yao Ma, Xiaorui Liu, Neil Shah, and Jiliang Tang. Is homophily a necessity for graph neural networks? In *ICLR*, 2022.

Jincheng Huang, Ping Li, Rui Huang, Na Chen, and Acong Zhang. Revisiting the role of heterophily in graph representation learning: An edge classification perspective. *ACM Transactions on Knowledge Discovery from Data*, 2023.

Xinyi Wu, Zhengdao Chen, William Wei Wang, and Ali Jadbabaie. A non-asymptotic analysis of oversmoothing in graph neural networks. In *ICLR*, 2022b.

Huayi Tang and Yong Liu. Towards understanding the generalization of graph neural networks. In *ICML*, 2023.

Hao Zhu and Piotr Koniusz. Simple spectral graph convolution. In *ICLR*, 2021.

Yifei Wang, Yisen Wang, Jiansheng Yang, and Zhouchen Lin. Dissecting the diffusion process in linear graph convolutional networks. In *NeurIPS*, 2021b.

Yong Lin, Lu Tan, Yifan Hao, Honam Wong, Hanze Dong, Weizhong Zhang, Yujiu Yang, and Tong Zhang. Spurious feature diversification improves out-of-distribution generalization. *arXiv preprint arXiv:2309.17230*, 2023.

13

Appendix

Table of Contents

A Additional Related Works   **15**
    A.1 Invariant Learning for OOD Generalization   15
    A.2 Graph-OOD Works Using VREx and Similar Variants   15
    A.3 Comparison with Existing Node-level OOD Generalization Works   16
    A.4 Graph-level OOD Generalization   17

B Additional Theoretical Results of the Covariate Shift Case   **17**
    B.1 Theoretical Model Setup of the Covariate Shift Case   17
    B.2 Theoretical Results of the Covariate Shift Case   17

C Detailed Experimental Setup   **18**
    C.1 Basic Settings   18
    C.2 Hyperparameter Settings of the Main OOD Generalization Results   18
    C.3 Details of the Toy Dataset   18

D Additional Experimental Results   **19**
    D.1 Excessive Alignment Leads to the Collapse of the Invariant Features.   19
    D.2 CIA-LRA Alleviates Representation Collapse Caused by Excessive Alignment .   19
    D.3 Effect of the Number of Hops for Localized Alignment   20
    D.4 Discussion and Validation of the Assumption on the Rate of Change of Causal and
          Spurious Features w.r.t Spatial Position   20
    D.5 Discussion and Validation of the Assumption on the Feature Distance and Neigh-
          borhood Label Distribution Discrepancy   21
    D.6 Validation of the True Feature Generation Depth   26
    D.7 Time Cost of CIA-LRA   29

E Detailed training procedure   **30**

F Additional Discussion of Theoretical Settings and Results   **30**
    F.1 Detailed Setup of the Theoretical Model in Section 2   30
    F.2 Discussion of the Structural Feature Considered in the Theoretical Model and
          Justification for the Choice of the Analyzed GNN   31
    F.3 Discussion of the Failure Solution for GNNs of VREx and IRMv1   31
    F.4 The Superiority of the Proposed CSBM-OOD   32
    F.5 Tightness of the Error Bound of Theorem 4.4   32

G Proofs of the Theoretical Results   **32**
    G.1 Proofs of the Concept Shift Case Presented in the Main Text   32
    G.2 Proof of the Covariate Shift Case   39
    G.3 Proof of Theorem 4.4   45

H Limitations   **48**

I Broader Impacts   **48**

J Compute Resources   **49**

14

A Additional Related Works

A.1 Invariant Learning for OOD Generalization

Invariant learning seeks to find stable features across multiple training environments to achieve OOD generalization [Arjovsky et al., 2020, Krueger et al., 2021, Bui et al., 2021, Rame et al., 2022, Shi et al., 2021, Wang et al., 2020, Mahajan et al., 2021, Wang et al., 2022a, Yi et al., 2022, Wang et al., 2022b]. IRM [Arjovsky et al., 2020] and VREx are two of the most well-known methods. The goal of IRM is to learn a representation that elicits a classifier to achieve optimality in all training environments. VREx [Krueger et al., 2021] reduces the variance of risks across training environments to improve robustness against distribution shifts. Mahajan et al. [2021] argues that the invariant features can also change across environments. Hence, they proposed MatchDG to align the representations of the so-called same "object" of different environments. To the best of our knowledge, we are the first work to theoretically analysis the limitations of IRM and VREx in OOD node classification problems.

It's worth emphasizing that although the CIA objective is similar to MatchDG [Mahajan et al., 2021], our extensions of MatchDG on graphs are non-trivial:

1) The extensions from MatchDG to CIA are mainly theoretical.

- We extend the idea of MatchDG to the node-level OOD task by providing a theoretical characterization of CIA's working mechanism on graphs (Theorem 3.1), revealing its superiority in node-level OOD scenarios for the first time.
- We establish the connection between node-level OOD error and two kinds of distributional shifts: (1) environmental node feature shifts and (2) the heterophilic neighborhood label distribution shifts (see Theorem 4.4), giving further explanation of CIA and CIA-LRA's performance gain in node-level OOD generalization.

2) The methodological extensions of CIA-LRA:

- It identifies the node pairs with significant differences in spurious features without using environment labels, providing a new perspective that the widely adopted environment partition paradigm Wu et al. [2021], Yu et al. [2023], Li et al. [2023b,a, 2022a], Liu et al. [2023] may not be necessary for node-level OOD generalization. One can remove spurious features by leveraging neighboring label distributions (analysis in Section 3.2), shedding light on the role of neighborhood label distribution as compensation for the absence of environment labels.
- It's the first node-level OOD method explicitly considering the OOD error caused by shifts in heterophilic neighborhood label distribution, (pointed out by term (c) in Theorem 4.4). Such shifts can be regarded as a kind of structural shift that has been first observed in Mao et al. [2023].
- It's the first node-level OOD method using homophilic neighborhood label distribution to reflect the find-grained distribution of the invariant features, avoiding the collapse of invariant features.

A.2 Graph-OOD Works Using VREx and Similar Variants

We summarize the graph-OOD works that include VREx or IGA as part of the training objectives below. Node-level works, which can be covered by our analysis:

- **EERM** [Wu et al., 2021], Equation (5):

$$\min_{\theta} \operatorname{Var}\left(\left\{ L\left(g_{w_k^*}(G), Y; \theta\right) : 1 \leq k \leq K \right\}\right) + \frac{\beta}{K} \sum_{k=1}^{K} L\left(g_{w_k^*}(G), Y; \theta\right), \quad (11)$$

  where $g_{w_k^*}(G)$ is the generated k-th environment.
- **INL** [Li et al., 2023a], Equation (8):

$$\mathbb{E}_{\text{e supp}(\mathcal{E}_{\text{infer}})} \mathcal{R}^e\left(f\left(G_v\right), y; \theta\right) + \lambda \operatorname{trace}\left(\operatorname{Var}_{\mathcal{E}_{\text{infer}}}\left(\nabla_{\theta} \mathcal{R}^e\right)\right), \quad (12)$$

  where $\mathcal{E}_{\text{infer}}$ is the inferred environment label.

15

- **FLOOD** [Liu et al., 2023], Equation (12):
$$\min_{\theta,\omega,\psi} \mathcal{L}_{\text{train}} = \mathcal{L}(\theta,\omega) + \alpha\mathcal{R}_{\text{V}-\text{REx}}(\omega,\psi) \tag{13}$$

- Zhang et al. [2021b] Equation (10):
$$\mathcal{L}_{\text{global}} = \sum_{e,e'\in\mathcal{E}} \left(\mathcal{L}_{\text{pred}}^{0,e} - \mathcal{L}_{\text{pred}}^{0,e'}\right)^2, \tag{14}$$

  as they claimed: "We note that minimizing the pair-wise distance of losses as defined in Equation 10 is equivalent to minimizing the variance of losses."

- **GLIDER** [Tian et al., 2024], Equation (9):
$$\min \mathbb{V}_e\left[\ell\left(f_c\left(g\left(G_v^e\right)\right), y_v^e\right)\right] + \alpha\mathbb{E}_e\left[\ell\left(f_c\left(g\left(G_v^e\right)\right), y_v^e\right)\right] \tag{15}$$

Graph-level works, which are not covered in our analysis:

- **LiSA** [Yu et al., 2023], Equation (15):
$$\min_f \mathcal{L}_{cls}\left(f, \{g_i^*\}_{i=1}^n\right) + \text{Var}_e\left(\mathcal{L}_{cls}\left(f, g_i^*\right)\right), i = 1\sim n, \tag{16}$$

  where g_i^* are the augmented training subgraphs of representing different environments.

- **G-splice** [Li et al., 2023b], Equation (6):
$$\psi^* := \underset{\psi}{\text{argmin}}\,\mathbb{E}_{(G,y)\sim\cup_{\varepsilon\in\{\mathcal{E}\cup\mathcal{E}_A\}}P_\varepsilon}\left[\ell\left(f_\psi(G), y\right)\right] + \gamma\,\text{Var}_{\varepsilon\in\{\mathcal{E}\cup\mathcal{E}_A\}}\left[\mathbb{E}_{(G,y)\sim P_\varepsilon}\ell\left(f_\psi(G), y\right)\right],$$
$$\tag{17}$$

  where $\mathcal{E}_A$ are the augmented environments.

- **GIL** [Li et al., 2022a], Equation (8):
$$\mathbb{E}_{e\in\text{supp}(\mathcal{E}_{infer})}\mathcal{R}^e(f(\mathrm{G}), \mathrm{Y}; \theta) + \lambda\,\text{trace}\left(\text{Var}_{\mathcal{E}_{\text{infer}}}\left(\nabla_\theta\mathcal{R}^e\right)\right), \tag{18}$$

  where $\mathcal{E}_{\text{infer}}$ is the inferred environment label.

A.3 Comparison with Existing Node-level OOD Generalization Works

Among the graph OOD methods, one line of research focuses on the node-level OOD generalization [Wu et al., 2021, Liu et al., 2023, Zhu et al., 2021, Li et al., 2023a, Xia et al., 2023]. We summarize the drawbacks of previous node-level OOD methods as follows. 1) it is hard for environment-inference-based methods to generate reliable environments. To generate environments, FLOOD [Liu et al., 2023] uses random data augmentation that lacks sufficient prior. EERM [Wu et al., 2021] generates environments by maximizing loss variance, which may not necessarily enlarge differences in spurious features across environments that have been proven to be crucial for invariant learning [Chen et al., 2023a]. Also, the adversarial learning of its environment generation process may lead to unstable performance (Table 2) and high training costs. INL [Li et al., 2023a] relies on an estimated invariant ego-graph of each node, whose quality could significantly affect performance. Moreover, all these methods need to manually specify the number of environments, which could be inaccurate. 2) previous node-level invariant learning objectives also have some limitations. For instance, Zhang et al. [2021b], Wu et al. [2021], Liu et al. [2023], Li et al. [2023a], Tian et al. [2024] use VREx Krueger et al. [2021] as their invariance regularization. However, we theoretically prove its potential failure on node-level OOD tasks in Section 2.2. SRGNN [Zhu et al., 2021] only aligns the marginal distribution $p(X)$ between the biased training distribution and the given unbiased distribution, which has been proved to have failure cases [Johansson et al., 2019]. 3) some work are based on intuitive guidelines and lacks theoretical guarantees on the OOD generalization performance [Liu et al., 2023, Zhu et al., 2021, Yang et al., 2023]. Our proposed CIA-LRA achieves invariant learning without complex environment inference that could be unstable through a representation alignment manner. Additionally, we provide theoretical guarantees for our invariant learning objective and empirically validate its working mechanism.

Comparison between our Theorem 2.3 and Theorem 1 of Wu et al. [2021] It's worth mentioning that Theorem 1 of Wu et al. [2021] proves $\min_\Theta \mathbb{V}_e[R(e)]$ will $\min I(y, e|z)$, where $q(z|x)$ is the induced distribution by encoder ϕ. This seems to conflict with Theorem 2.3. This is because the upper bound derived in Wu et al. [2021] $I(y, e|z) \leq D_{\text{KL}}(q(y|z)\|\mathbb{E}_e[q(y|z)]) \leq \mathbb{V}_e[R(e)]$ is not tight. Thus, minimizing the variance of loss across training environments does not necessarily minimize mutual information between the label and the environment given the learned representation.

16

A.4 Graph-level OOD Generalization

There has been a substantial amount of work focusing on the OOD generalization problem on graphs. The vast majority have centered on graph classification tasks[Chen et al., 2022, Wu et al., 2022a, Li et al., 2022a, 2023b, Yang et al., 2022, Yu et al., 2023, Chen et al., 2023a, Buffelli et al., 2022, Jia et al., 2023, Li et al., 2022b, Chen et al., 2023b, 2024, Gui et al., 2024]. Most works aim at identifying the invariant subgraph of a whole graph through specific regularization so that the model can use it when inference. Compared to node-level OOD generalization, the graph-level one is more akin to traditional OOD generalization, as the individual samples (graphs) are independently distributed. We focus on the more challenging node-level OOD generalization in this work.

B Additional Theoretical Results of the Covariate Shift Case

B.1 Theoretical Model Setup of the Covariate Shift Case

In this section, we will extend our theoretical model in the main text to the covariate shift setting. The causal graph of the covariate shift is shown in Figure 1b. For the covariate shift setting, spurious features are independent of Y, while X changes with environment e. Thus we can model the data generation process for environment e as

$$Y^e = (\tilde{A}^e)^k X_1 + n_1, \quad X_2^e = n_2 + \epsilon^e, \tag{19}$$

where the definition of n_1 and n_2 are the same as Section 2, ϵ^e represents environmental spurious features. ϵ_i^e (each dimension of ϵ^e) is a random variable that are independent for $i = 1, ..., N^e$. We assume the intra-environment expectation of the environment spurious variable is $\mathbb{E}_{\epsilon_i \sim p_e}[\epsilon_i] = \mu^e \in \mathbb{R}$ since spurious features are consistent in a certain environment. We further assume the cross-environment expectation $\mathbb{E}_e[\epsilon^e] = \mathbf{0}$ and cross-environment variance $\mathbb{V}_e[\epsilon_i^e] = \sigma^2$, $i = 1, ..., N^e$ for simplicity. This is consistent with the covariate shift case that $p(X)$ can arbitrarily change across different domains, and the support set of X may vary. Also, we require $L \geq k$ to ensure the GNN has enough capacity to learn the causal representations.

B.2 Theoretical Results of the Covariate Shift Case

Now we will present similar conclusions as the concept shift case. Even if VREx and IRMv1 can successfully capture invariant features in the non-graph task, they induce a model that uses spurious features. Still, CIA can learn invariant representations under covariate shift.

Proposition B.1. *(VREx and IRMv1 learn invariant features for non-graph tasks under covariate shift, proof is in Appendix G.2.1) For the non-graph version of the SCM in Equation (19),*

$$Y^e = X_1 + n_1, \ X_2^e = n_2 + \epsilon^e, \tag{20}$$

Optimizing VREx $\min_\Theta \mathcal{L}_{VREx} = \mathbb{V}_e[R(e)]$ and IRMv1 $\min_\Theta \mathcal{L}_{IRMv1} = \mathbb{E}_e[\|\nabla_{w|w=1.0} R(e)\|^2]$ will learn invariant features when using a 1-layer linear network: $f(X) = \theta_1 X_1 + \theta_2 X_2$.

Proposition B.2. *(VREx will use spurious features on graphs under covariate shift) Under the SCM of Equation (19), the objective $\min_\Theta \mathbb{V}_e[R(e)]$ has non-unique solutions for parameters of the GNN (3) when part of the model parameters $\{\theta_1^{1(l)}, \theta_1^{2(l)}, \theta_2^{1(l)}, \theta_2^{2(l)}\}$ take the values*

$$\Theta_0 = \begin{cases} \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s+1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s, L-s-1, ..., 1 \\ \theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-1, ..., 1 \end{cases}, \tag{21}$$

$0 < s < L$ is some positive integer. Specifically, the VREx solutions of θ_1 and θ_2 are the sets of solutions of the cubic equation, some of which are spurious solutions that $\theta_2 \neq 0$ (although $\theta_2 = 0$ is indeed one of the solutions, VREx is not guaranteed to reach this solution):

$$\begin{cases} c_1\sigma^2(2\theta_1\theta_2 + (\theta_2)^2 - 2c_2\sigma^2\theta_2) + c_3\theta_2 - \mathbb{E}_e[N^e]c_1\sigma^2\theta_1\theta_2 + \mathbb{E}_e[N^e]c_2\sigma^2\theta_2 = 0 \\ \left[c_3\theta_2 - \mathbb{E}_e[N^e]c_1\sigma^2\theta_1\theta_2 + \mathbb{E}_e[N^e]c_2\sigma^2\theta_2\right]c_4 - c_5(\theta_2)^2 = 0 \end{cases}. \tag{22}$$

where $c_1 = \mathbb{E}[(\tilde{A}^{e^s} X_1)^\top (\tilde{A}^{e^s} X_1)]$, $c_2 = \mathbb{E}[(\tilde{A}^{e^s} X_1)^\top (\tilde{A}^{e^k} X_1)]$, $c_3 = \mathbb{E}_e[\epsilon^{e\top} \epsilon^e \epsilon^{e\top} (\tilde{A}^{e^s} X_1)]\sigma^2$, $c_4 = \mathbb{E}_e\left[((\tilde{A}^e)^k X_1)^\top \mathbf{1_{N^e}}\right]\sigma^2$, $c_5 = \mathbb{E}_e\left[N^e\left(tr((\tilde{A}^{e^k})^\top \tilde{A}^{e^k}) + N^e(1 + \sigma^2)\right)\right]$.

17

Proposition B.3. *(IRMv1 will use spurious features on graphs under covariate shift) Under the SCM of Equation (19), there exists $s \in \mathbb{N}^+$ that satisfies $0 < s < L$ and $s \neq k$ such that optimizing the IRMv1 objective $\min_\Theta \mathbb{E}_e[\|\nabla_{w|w=1.0} R(e)\|^2]$ will not lead to the invariant solution $\theta_2 = 0$ for parameters of the GNN (3) when $\{\theta_1^{1(l)}, \theta_1^{2(l)}, \theta_2^{1(l)}, \theta_2^{2(l)}\}$ take the special solution:*

$$\Theta_0 = \begin{cases} \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s+1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s, L-s-1, ..., 1 \\ \theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-1, ..., 1 \end{cases} . \tag{23}$$

Proposition B.4. *Optimizing the CIA objective will lead to the optimal solution Θ^*:*

$$\begin{cases} \theta_1 = 1 \\ \theta_2 = 0 \\ \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-k+1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-k, L-k-1, ..., 1 \end{cases} . \tag{24}$$

C Detailed Experimental Setup

C.1 Basic Settings

All experimental results were averaged over three runs with different random seeds. Following Gui et al. [2022], we use an OOD validation set for model selection.

GAT experiments. For experiments on GAT, we adopt the learning rate of 0.01 for Arxiv, 0.001 for Cora, 0.003 for CBAS, and 0.1 for WebKB. The reason why we didn't use the default learning rate in Gui et al. [2022] is that since the original GOOD benchmark didn't implement GAT, so we chose to tune a learning rate for adapting GAT to reach a decent performance. The settings of batch size, training epochs, weight decay, and dropout follow Gui et al. [2022].

GCN experiments. For experiments on GCN, we follow the default settings of batch size, training epochs, learning rate, weight decay, and dropout provided by Gui et al. [2022].

It's worth mentioning that we choose different normalization strategies for the invariant edge mask of CIA-LRA to achieve better performance for GCN. In Equation (8), we use Sigmoid as normalization. However, we find it is better to use a Min-Max normalization for GCN on some of the datasets. Specifically, for experiments on GCN, we use Sigmoid normalization for CBAS, Arxiv (time concept); Sigmoid for training and Min-Max for testing for WebKB (covariate); Min-Max normalization for the other dataset splits. For GAT, we use Sigmoid during training and testing for all datasets.

C.2 Hyperparameter Settings of the Main OOD Generalization Results

Most hyperparameter settings are adopted from Gui et al. [2022], except that for EERM we reduce the number of generated environments from 10 to 7 and reduce the number of adversarial steps from 5 to 1 for memory and time complexity concerns. For each method, we conduct a grid search for about 3∼7 values of each hyperparameter. The hyperparameter search space is presented in Table 5.

C.3 Details of the Toy Dataset

Now we introduce the setting of the toy synthetic task of Figure 3. The synthetic dataset consists of four classes. Each node has a 4-dim node feature. The first/last two dimensions correspond to the invariant/spurious feature for each of the four classes, as shown in Table 6a and 6b. We artificially create both concept shift and covariate shift in this dataset.

To explicitly disentangle the learned invariant and the spurious components for quantitative analysis, we employ a 1-layer GCN. We take the output of the first/last two dimensions of the weight matrix as the invariant/spurious representation.

18

Table 5: Hyperparameter setting of the experiments.

Algorithm	Search Space
IRM	λ= 0.1, 1, 10, 100
VREx	λ=1, 10, 100, 1000
GroupDRO	λ=0.001, 0.01, 0.1
Deep Coral	λ=0.01, 0.1, 1
IGA	λ=0.1, 1, 10, 100
Mixup	α=0.4, 1.0, 2.0
EERM	β=0.5, 1, 3 number of generated environments k=7 adversarial training steps t=1 numbers of nodes for each node should be modified the link with s=5 subgraph generator learning rate r=0.0001, 0.001, 0.005, 0.01
SRGNN	0.000001, 0.00001, 0.0001
GTrans	feature learning rate r_1=0.000005, 0.00001, 0.0001 structure learning rate r_2=0.01, 0.1 δ optimization steps t=5, 10
CIA	λ=0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1 (GCN) λ=0.0001, 0.0005, 0.001, 0.01, 0.05 (GAT)
CIA-LRA	λ= 0.001, 0.005, 0.01, 0.05, 0.1 (GCN) edge mask GNN learning rate r=0.001 (GCN + Arxiv, Cora, CBAS); $r = 0.1$ (GCN + WebKB) λ= 0.005, 0.05, 0.1, 0.5 (GAT) edge mask GNN learning rate r=0.0001 (GAT) hops t=2, 3, 4, 5, 6

Table 6: Information of the toy dataset of Figure 3.

(a) The invariant part (the first two dimensions) of the node features in the synthetic dataset.

class 0	class 1	class 2	class 3
$\mathcal{N}((1,1),I)$	$\mathcal{N}((-1,1),I)$	$\mathcal{N}((-1,-1),I)$	$\mathcal{N}((1,-1),I)$

(b) The spurious part (the last two dimensions) of the nodes features in the synthetic dataset.

environment	concept shift				covariate shift
	class 0	class 1	class 2	class 3	
training 1	$\mathcal{N}((3,3),I)$	$\mathcal{N}((-3,3),I)$	$\mathcal{N}((-3,-3),I)$	$\mathcal{N}((3,-3),I)$	$\mathcal{N}((2,2),I)$
training 2	$\mathcal{N}((-3,3),I)$	$\mathcal{N}((-3,-3),I)$	$\mathcal{N}((3,-3),I)$	$\mathcal{N}((3,3),I)$	$\mathcal{N}((4,4),I)$
test	$\mathcal{N}((-3,-3),I)$	$\mathcal{N}((3,-3),I)$	$\mathcal{N}((3,3),I)$	$\mathcal{N}((-3,3),I)$	$\mathcal{N}((6,6),I)$

D   Additional Experimental Results

D.1   Excessive Alignment Leads to the Collapse of the Invariant Features.

One may wonder why CIA underperforms CIA-LRA even if CIA uses the ground truth environment labels. In this section, we will show that CIA may suffer excessive alignment, which will lead to the collapse of the learned invariant features and consequently hurt generalization. We use the intra-class variance of the representation corresponding to invariant features (averaged over all classes) to measure the degree of collapse of invariant features. Base on this measurement, the excessive alignment can be caused by:

1) Using a that is too large. Evidence: on the toy dataset of Figure 3, a larger λ leads to smaller intra-class variance of invariant representations. We also compute the intra-class variance of invariant representations at epoch 50 on the toy dataset,

2) Aligning the representations of too many nodes. Evidence: we show that aligning fewer node pairs can alleviate the collapse of invariant representation. By modifying CIA to align local pairs (same-class, different-environment nodes within 1 hop), termed "CIA-local", the results in Table 7b show that when by aligning local pairs instead of all pairs, CIA-local avoids the collapse that CIA suffers and achieves better performance.

D.2   CIA-LRA Alleviates Representation Collapse Caused by Excessive Alignment

In Figure 5, we illustrate the impact of λ on OOD accuracy. Both CIA and CIA-LRA experience a performance decline at $\lambda = 0.5$, indicating that excessive alignment can hinder generalization. Furthermore, CIA shows an earlier and more pronounced performance drop than CIA-LRA. This suggests that the CIA-LRA method mitigates representation collapse by aligning fewer pairs and selectively focusing on pairs with smaller differences in invariant features.

19

Table 7: Experimental evidence of the factors that can cause the collapse of the learned invariant representations.

(a) **Using a that is too large can cause the collapse.** The intra-class variance of invariant representations on the toy dataset of Figure 3 at epoch 50.

CIA	$\lambda = 0.05$	$\lambda = 0.1$	$\lambda = 0.5$
variance of the invariant representation	0.061	0.039	0.011

(b) **Aligning the representations of too many nodes can cause the collapse.** Accuracy and the variance of the invariant representations on the toy dataset of Figure 3 at epoch 200.

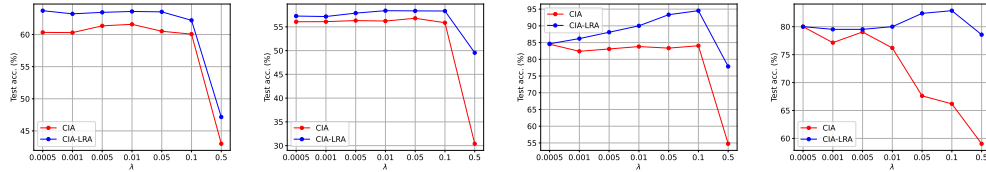
	CIA	CIA-local
Concept shift, accuracy	0.253	0.354
Concept shift, variance of the invariant representation	0.0003	0.2327
Concept shift, accuracy	0.250	0.312
Concept shift, variance of the invariant representation	0.0002	0.1699

(a) Cora degree concept (b) Cora degree covariate (c) CBAS color concept (d) CBAS color covariate

Figure 5: The effect of λ on OOD accuracy. CIA exhibits an earlier and more severe performance drop than CIA-LRA, demonstrating that CIA-LRA can alleviate the feature collapse caused by excessive alignment.

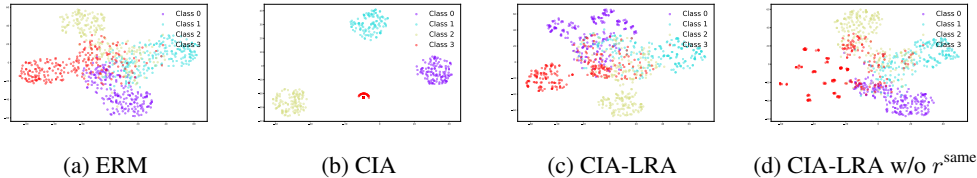
The role of CIA-LRA in alleviating the collapse of the invariant features can also be reflected in Figure 6, in which the representation learned by CIA collapsed to a compact region. However, CIA-LRA does not exhibit such collapse, maintaining the diversity of the causal representation.

(a) ERM (b) CIA (c) CIA-LRA (d) CIA-LRA w/o r^{same}

Figure 6: Visualization of the learned representations at epoch 100 on the toy dataset (concept shift). Classes are distinguished by color. $\lambda = 0.5$ for CIA and CIA-LRA.

D.3 Effect of the Number of Hops for Localized Alignment

In Figure 7, we plot the OOD accuracy curve of CIA-LRA against the number of hops t for localized alignment (with $\lambda = 0.05$). CIA-LRA achieves optimal performance within a local range of 6 to 10 hops. Performance is notably lower at smaller hops ($t = 2$), due to limited regularization from aligning only a few pairs of representations. As t increases, performance gains diminish and can even degrade, particularly on the CBAS color covariate. This underscores the importance of localized alignment: optimal OOD performance is attained by aligning nodes within about 10 hops. Extending the alignment range further does not enhance performance significantly and may lead to performance drops and higher computational costs. These findings support the hypothesis in Appendix D.4 that invariant features distant on the graph differ substantially, and their alignment could induce invariant feature collapse, leading to a suboptimal generalization performance.

D.4 Discussion and Validation of the Assumption on the Rate of Change of Causal and Spurious Features w.r.t Spatial Position

To verify the intuition presented in Section 3.2 that spurious features exhibit larger changes within a local range (about 5 to 10 hops) on a graph compared to invariant features, we conduct experiments

20

(a) Cora degree concept   (b) Cora degree covariate   (c) CBAS color concept   (d) CBAS color covariate

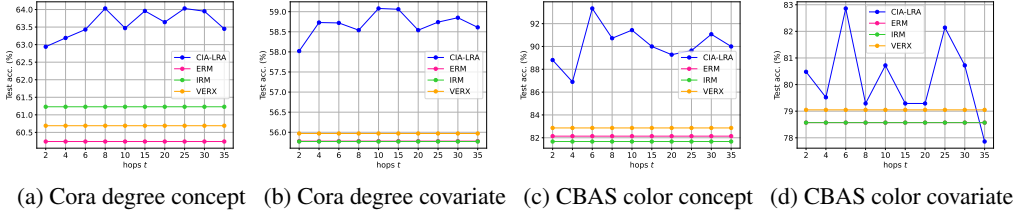



Figure 7: The effect of the number of hops t for localized alignment on OOD accuracy. Too small t will lead to suboptimal performance. Too large t brings limited performance gain or even deteriorates the performance.

on real-world datasets Arxiv and Cora. To extract invariant features, we use a pre-trained VREx model and take the output of the last layer as invariant features[8]. To obtain spurious features, we train an ERM model to predict the environment label and take the output of the last layer as spurious features. For each class, we randomly sample 10 nodes and generate corresponding 10 paths using Breadth-First Search (BFS). We extract invariant and spurious features of the nodes on each path and plot the L-2 distances between the node representations on the paths and the starting node. The results of Cora are in Figure 8 and 9, and the results of Arxiv are in Figure 10 and 11. We chose some of the classes to avoid excessive paper length; the results for the other classes are similar.

We observe that: despite the curve's slight fluctuations, the invariant feature difference shows a clear positive correlation with the distance from the starting point. Specifically, within about 5∼10 hops, the changes of spurious features grow more rapidly than those in invariant ones. This insight led us to align the representations of adjacent nodes to better eliminate spurious features and avoid the collapse of the invariant features. This also explains why we add a weighting term $d(i, j)$ in our loss function to assign smaller weight node pairs farther apart. Additional experimental evidence supporting the importance of localized alignment is in Appendix D.3, which shows that alignment over a large range may lead to suboptimal performance and increasing computational costs.

This assumption aligns with those adopted in a series of previous works on causality and invariant learning [Chen et al., 2022, Burshtein et al., 1992, Schölkopf, 2022, Schölkopf et al., 2021]. These works assume that invariant features are better clustered than spurious features. In the node-level graph OOD scenario, we observe this phenomenon primarily within local parts of a graph. In some cases, when two nodes are too far apart, their invariant features can vary more than the spurious features, as seen in Figure 11 (a) path 1,2,4,6,9 and 10. Therefore, matching the representations in a local region helps alleviate the invariant feature collapse problem.

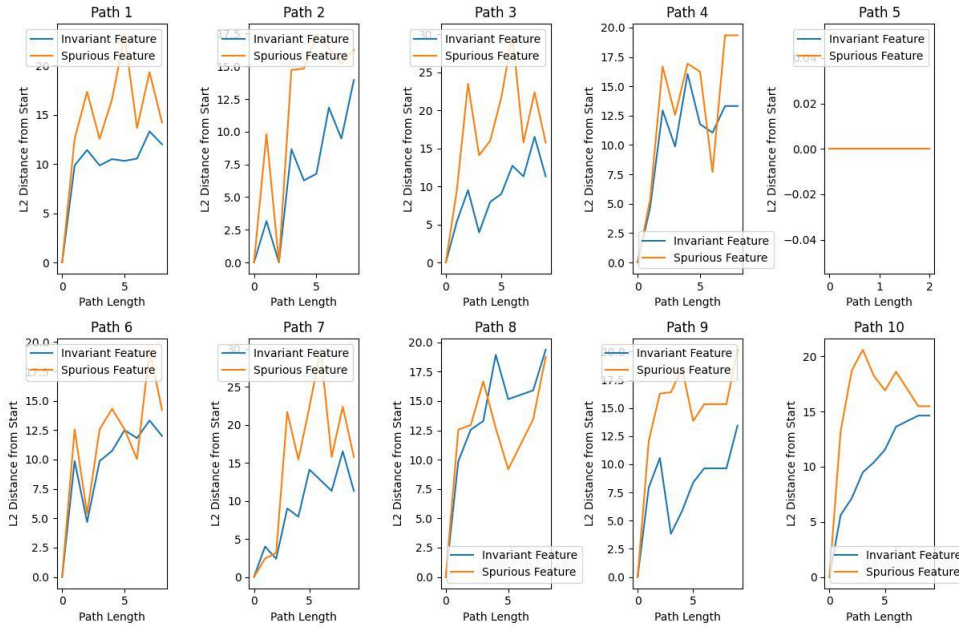
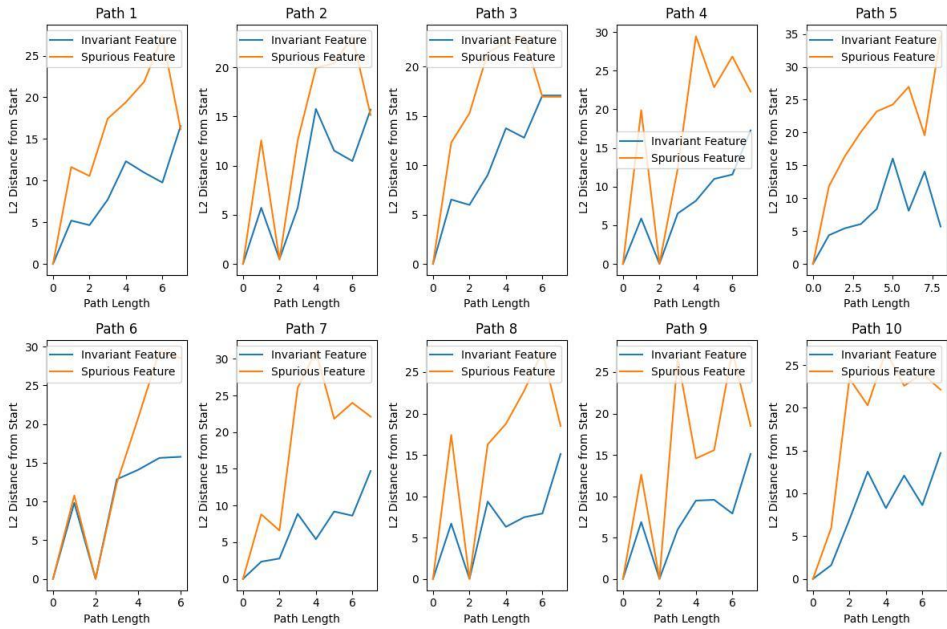
D.5   Discussion and Validation of the Assumption on the Feature Distance and Neighborhood Label Distribution Discrepancy

D.5.1   Heterophilic Neighborhood Labels Distribution Reflect Spurious Feature Distribution

In this section, we will empirically validate the key intuition of CIA-LRA: the label distribution of the neighbors from different classes (which we call *Heterophilic Neighborhood Label Distribution*, HeteNLD) reflects the spurious representation of the centered node. In node-level OOD scenarios, the distributional shifts of spurious features originate from two main sources: (1) the shifts in spurious node features associated with environments, and (2) the shifts in *Neighborhood Label Distribution* (NLD), which affects the aggregated representation of the centered node. The first type of spurious feature is analogous to those defined in Computer Vision (CV) OOD domains, while the second type is specific to graph structures. The NLD shift is a more general instance of the graph heterophily problem [Ma et al., 2022, Huang et al., 2023, Mao et al., 2023], where changes in the ratio of homophilic neighbors from training to test graphs can degrade performance. This occurs because the changes in the homophilic ratio lead to the distributional shift in the aggregated representation of the same-class nodes. Most previous methods [Ma et al., 2022, Huang et al., 2023, Mao et al., 2023]

---

[8]though we reveal in our theory that VREx may rely on spurious features, we still use VREx here to approximately extract invariant features as many previous graph OOD works have done since VREx already demonstrated some advantages in their works

21

(a) class 16 of Cora

(b) class 17 of Cora

Figure 8: Visualization of the rate of change of invariant features and spurious features on Cora (part 1).

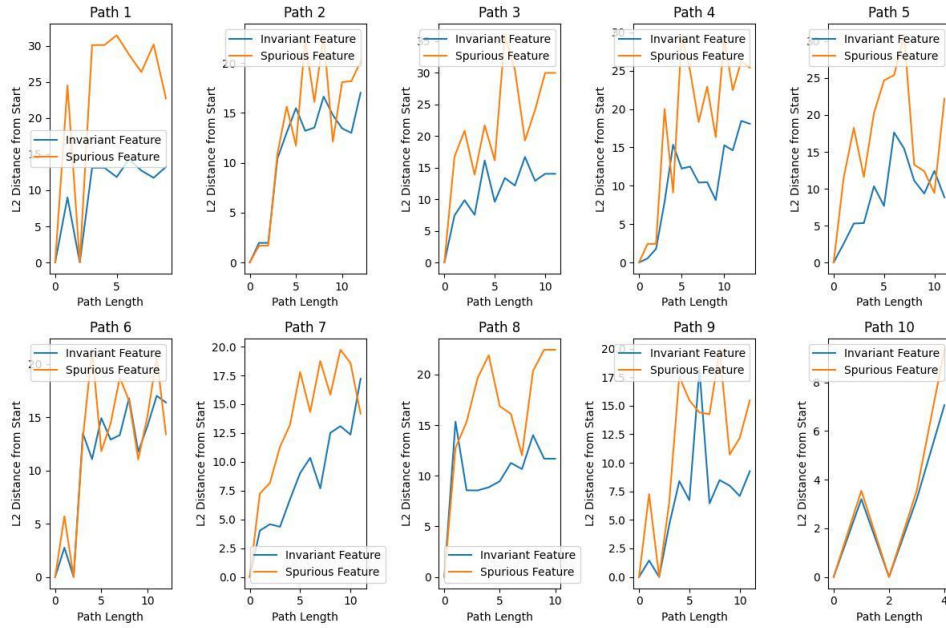
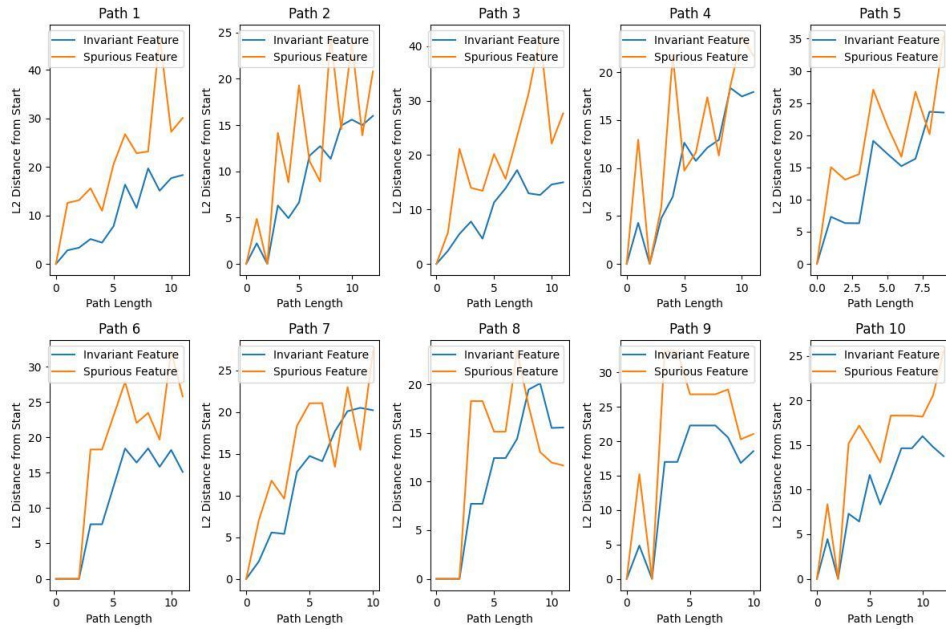
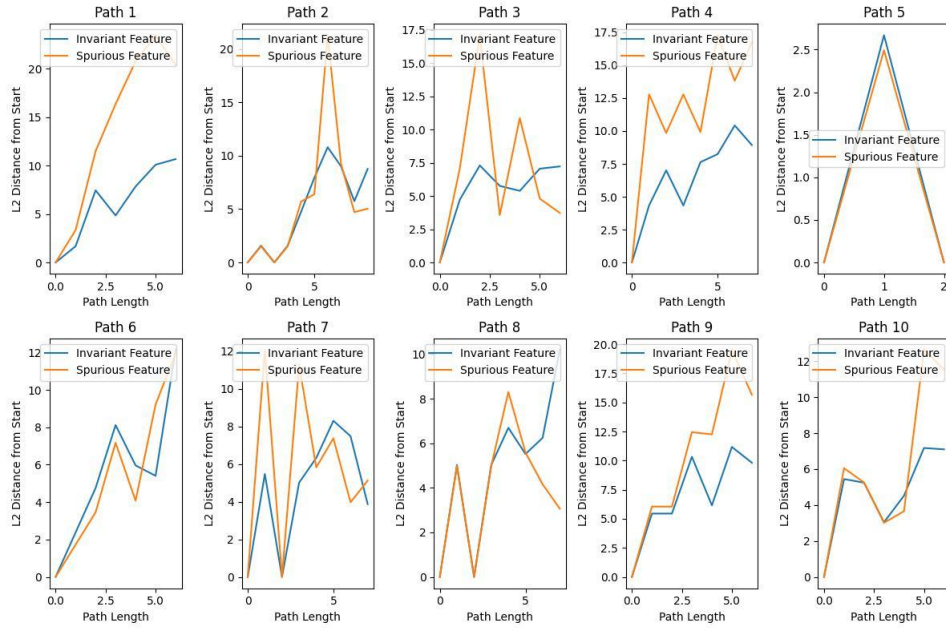
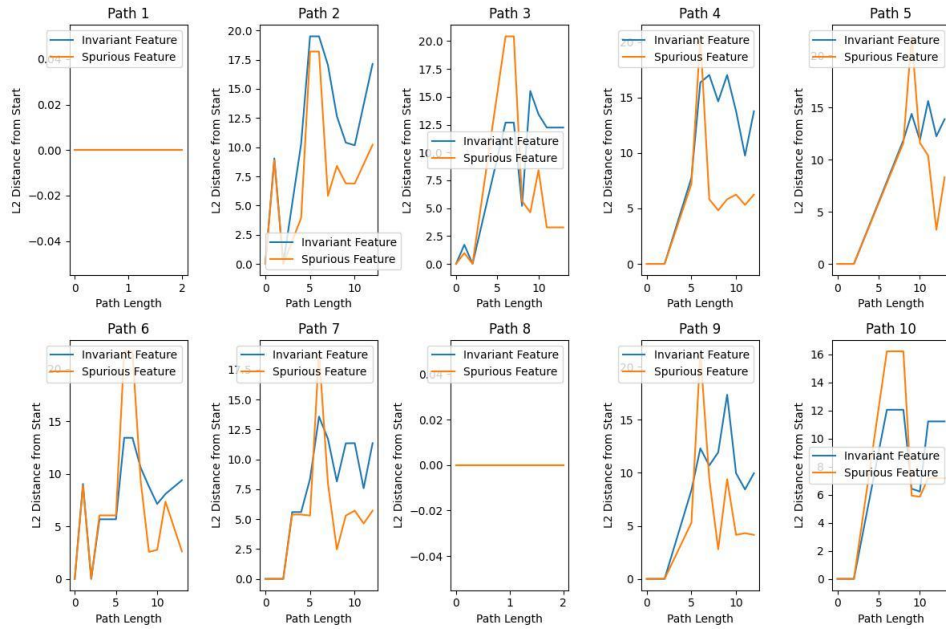
22

(a) class 39 of Cora

(b) class 41 of Cora

Figure 9: Visualization of the rate of change of invariant features and spurious features on Cora (part 2).

23

(a) class 25 of Arxiv

(b) class 29 of Arxiv

Figure 10: Visualization of the rate of change of invariant features and spurious features on Arxiv (part 1).

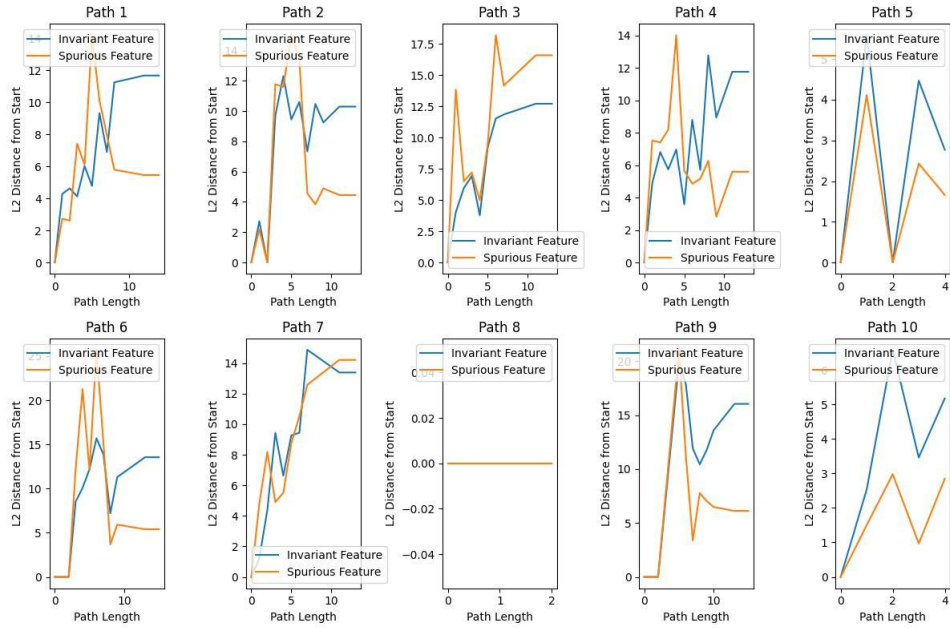
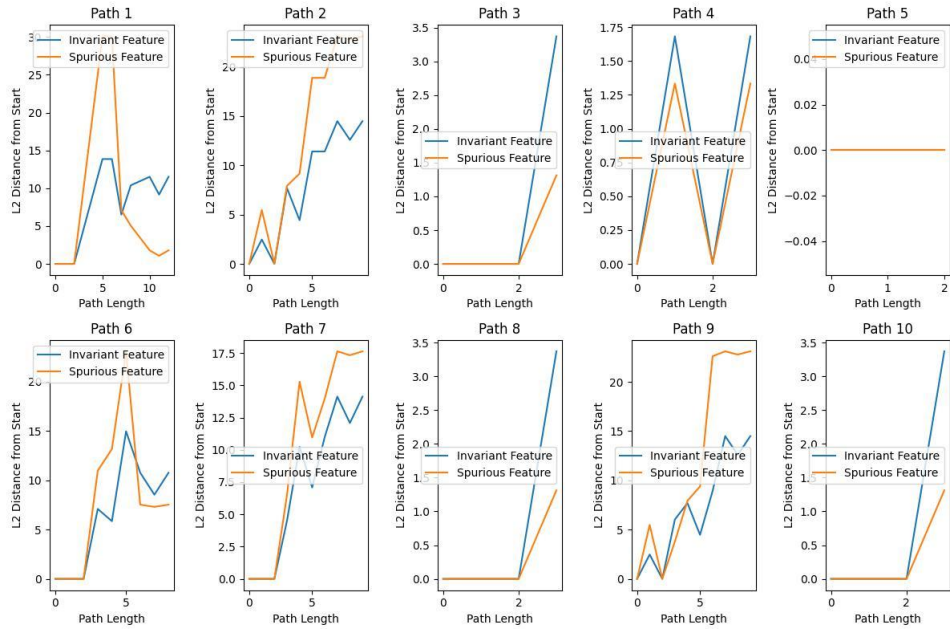
24

(a) class 13 of Arxiv

(b) class 17 of Arxiv

Figure 11: Visualization of the rate of change of invariant features and spurious features on Arxiv (part 2).

25

only focus on the binary-classification setting, where changes in the homophilic neighbor ratio are equivalent to changes in the heterophilic neighbor ratio. However, we consider the more general multi-classification tasks. Therefore, we propose to use HeteNLD as a measurement, considering every class different from the central class and using their distribution to reflect shifts in the aggregated representation. Although the ratio of homophilic neighbors also affects environmental spurious features and NLD, it affects the invariant representation as well. Assigning larger weights to the pair with significant differences in the ratio of homophilic neighbors will simultaneously eliminate environmental spurious features and learn a collapsed invariant representation. As evidenced in Table 4, moving the $r^{\text{same}}(c)_{i,j}$ to the numerator of Equation (9) will lead to a significant performance decrease. Hence we use $\frac{1}{r^{\text{same}}(c)_{i,j}}$ instead of $r^{\text{same}}(c)_{i,j}$ in w_{ij}.

In the following part, we will empirically validate our intuition that HeteNLD can reflect the two spurious representation distributions on *concept shift*, where $p(Y|X)$ varies across environments, and *covariate shift*, where $p(X)$ changes with environments, respectively. We will show that HeteNLD affects the spurious features of the centered node in different manners under concept shift and covariate shift.

Covariate shift. For covariate shifts on graphs, since spurious features are not necessarily correlated with labels, the environmental spurious features cannot be reflected by HeteNLD. However, we can still measure how HeteNLD affects the aggregated neighborhood representation. To obtain neighborhood representation, we train a 1-layer GCN that aggregates neighboring features and discards the features of the centered node. We hope to observe whether the gap of HeteNLD accurately reflects the distance of neighborhood representation. To ensure that the discrepancy in the aggregated neighboring feature is caused solely by heterophilic neighbors, we only use point pairs with the same number of homophilic neighbors. Specifically, we compute the L-2 distance between the neighborhood representations of two nodes with the same number of class-same neighbors, and plot its trend w.r.t. the distance of HeteNLD (according to the definition of $r_{i,j}^{\text{diff}}$ in Equation (9), except that we didn't normalize by the node degree here). We run experiments on Cora to verify this. We evaluate on both *word* shifts (node feature shifts) and *degree* (graph structure shifts) for a comprehensive understanding. We show the results of the first 30 classes of Cora. **The results in Figure 12 and 13 show a clear positive correlation between the neighborhood representation distance and HeteNLD discrepancy under covariate shifts, indicating HeteNLD discrepancy can reflect the distance of the aggregated representation.**

Concept shift. As for concept shift, spurious features are correlated with labels, thus the label of a node contains information about spurious features correlated with this class. Hence, by observing HeteNLD, we can measure the distribution of the spurious feature. For concept shift, we train a GNN to predict environment labels to obtain spurious representations. **Table 14 and 15 also show a clear positive correlation between spurious feature distance and HeteNLD discrepancy on concept shift, indicating that HeteNLD discrepancy can reflect the distance of the environmental spurious features.**

D.5.2 Homophilic Neighboring Labels Reflect Invariant Feature Distribution

Now will validate that the ratio of the same-class neighbors reflects the aggregated invariant representation. We use VREx to approximately extract invariant features and compute their distance w.r.t. the discrepancy of the ratio of the same-class neighbors. We evaluate on 4 splits of Cora: *word*+**covariate**, *word*+**concept**, *degree*+**covariate** and *degree*+**concept**. For each data split, we randomly choose 5 classes with sufficiently large differences in homophilic neighbor ratios for visualization. **The results in Figure 16 also show a positive correlation trend between the distance of the invariant representations and the difference in the ratio of same-class neighbors, indicating the latter can reflect the former.**

D.6 Validation of the True Feature Generation Depth

For the theoretical model in Section 2, we assume that the number of layers of the GNN L is greater than the depth of the causal pattern k. In this section, we empirically verify how large k really is on real-world datasets. Specifically, we use GCN with different layers to predict the ground-truth label Y on Cora and Arxiv datasets respectively (results are in Table 9a and 9b). As mentioned above, since a GCN with L layers will aggregate features from L-hop neighbors for prediction, if the depth

26

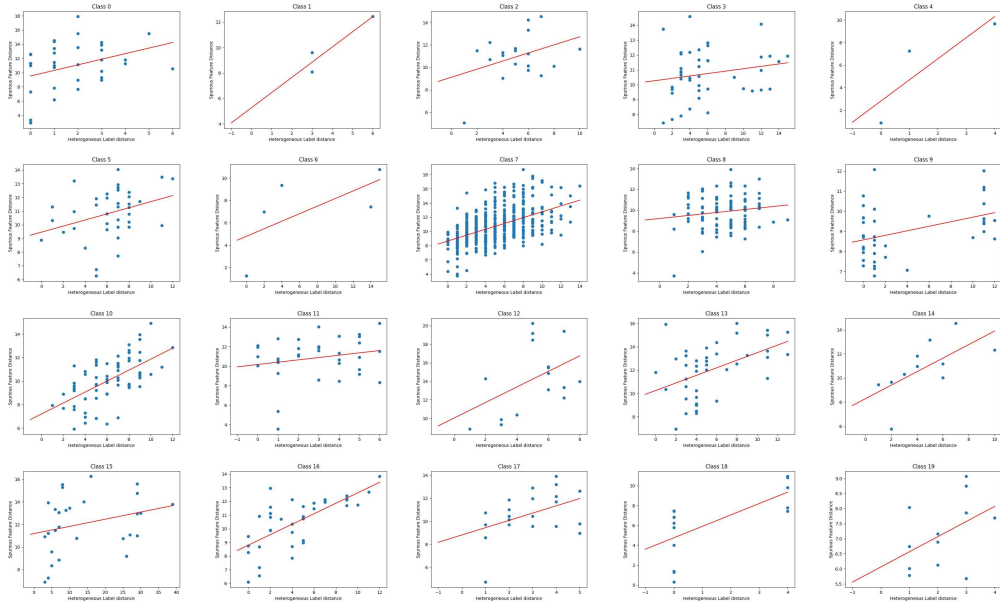



Figure 12: The relationship between the distance of the aggregated neighborhood representation and distance of HeteNLD on Cora *word* domain, **covariate shift**. Each sub-figure is a class, and each dot in the figure represents a node pair in the graph. The red line is obtained by linear regression. The positive correlation is clear.

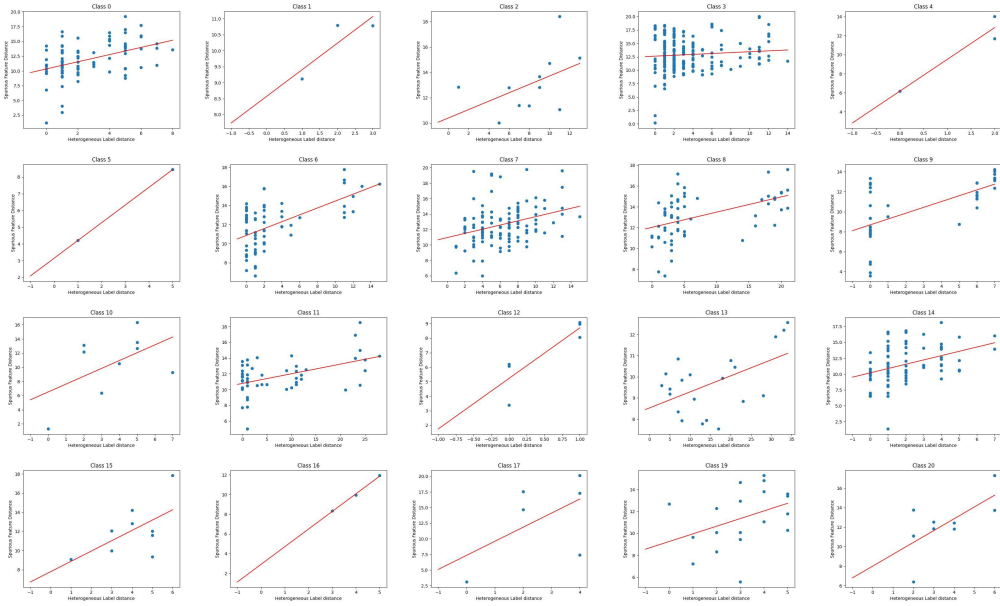



Figure 13: The relationship between the distance of the aggregated neighborhood representation and distance of HeteNLD on Cora *degree* domain, **covariate shift**. The positive correlation is clear.

27

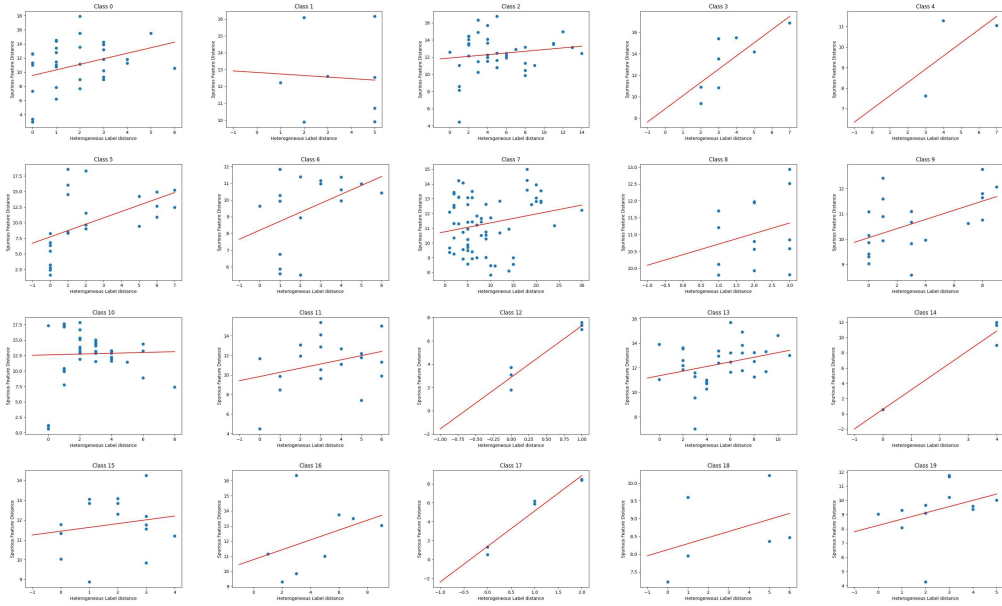



Figure 14: The relationship between the distance of environmental spurious features and distance of HeteNLD on Cora *word*, **concept shift**. The positive correlation holds for most classes.

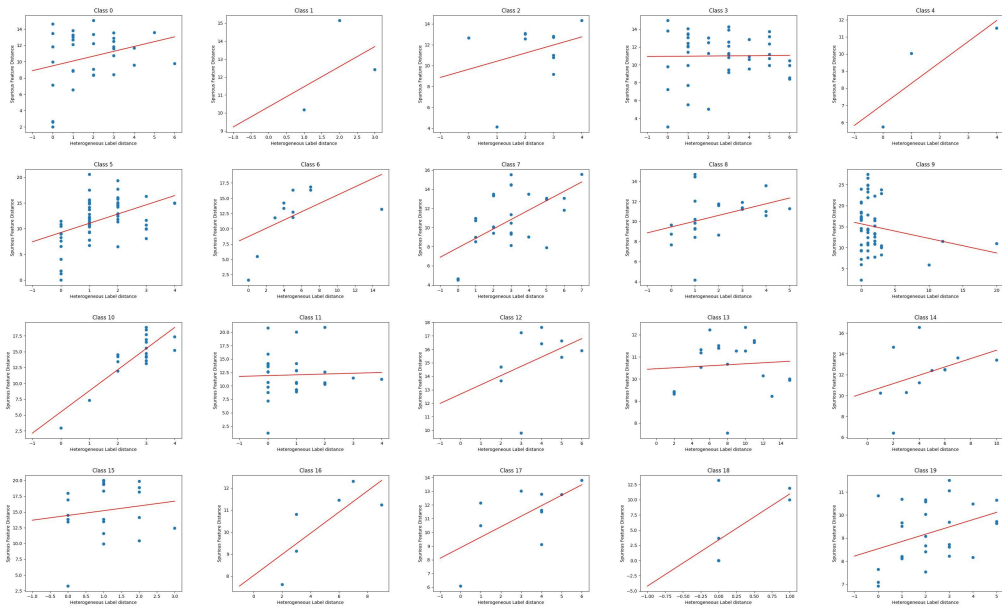



Figure 15: The relationship between the distance of environmental spurious features and distance of HeteNLD on Cora *degree*, **concept shift**. The positive correlation holds for most classes.

28

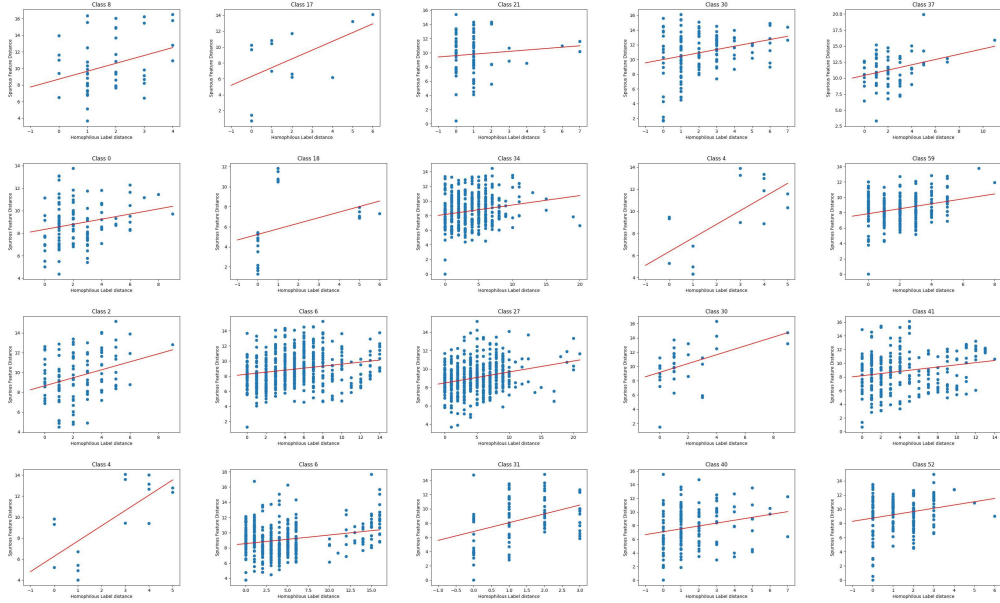



Figure 16: The relationship between the distance of invariant representations and discrepancy in the same-class neighbor ratio (the ratios in the figure are multiplied by node degree) on Cora *degree*, **concept shift**. Line 1 to 4 are results of Cora *word*+**covariate**, *word*+**concept**, *degree*+**covariate** and *degree*+**concept**, respectively. Each subgraph marks a class, and each point in the subfigure represents a node pair. There is a positive correlation between the invariant feature distance and the difference in neighboring label ratio of the same class as the centered node.

Table 8: Time cost (seconds) to achieve optimal test performance on Arxiv using GAT on a single RTX 3090 GPU.

	ERM	Coral	Mixup	EERM	SRGNN	GTrans	CIT	CIA-LRA (6 hops)
Arxiv degree covariate	74	551	758	OOM	34887	OOM	OOM	1248
Arxiv degree concept	30	360	747	OOM	3960	OOM	OOM	1132
Arxiv time covariate	46	246	1207	OOM	1993	OOM	OOM	292
Arxiv time concept	440	1481	272	OOM	11628	OOM	OOM	989

of the GCN is equal to the true generation depth, then the performance should be close to optimal. Therefore, we use the layer number that yields the optimal empirical performance (denoted as L^*) to approximate k. **We find that the $L^* \leq 4$ in most cases.** This indicates that our assumptions $L \leq k$ hold easily.

D.7 Time Cost of CIA-LRA

To show the running time of CIA-LRA, we show the time cost to reach the best test accuracy on our largest dataset Arxiv (with 50k 60k nodes). The results are in Table 8 below. The time cost of CIA-LRA is comparable to baseline methods.

(a) OOD accuracy on causal prediction (%) of GCN with different numbers of layers on Arxiv.

Dataset	Shift	$l = 2$	$L = 3$	$L = 4$	L=5
Arxiv (degree)	covariate	57.28(0.09)	58.92(0.14)	**60.18(0.41)**	60.17(0.12)
	concept	63.32(0.19)	62.92(0.21)	**65.41(0.13)**	63.93(0.58)
Arxiv (time)	covariate	71.17(0.21)	70.98(0.20)	**71.71(0.21)**	70.84(0.11)
	concept	65.14(0.12)	67.36(0.07)	65.20(0.26)	**67.49(0.05)**

(b) OOD accuracy (%) of GCN with different numbers of layers on Cora.

Dataset	Shift	$L = 1$	$l = 2$	$L = 3$	$L = 4$
Cora (degree)	covariate	**59.04(0.15)**	58.44(0.44)	55.78(0.52)	55.15(0.24)
	concept	**62.88(0.34)**	61.53(0.48)	60.24(0.40)	60.51(0.17)
Cora (word)	covariate	64.05(0.18)	**65.81(0.12)**	65.07(0.52)	64.58(0.10)
	concept	64.32(0.15)	**64.85(0.10)**	64.61(0.11)	64.16(0.23)

29

E Detailed training procedure

Table 1 shows the detailed training procedure (pseudo code) of CIA-LRA. We use the same GNN encoder for the invariant subgraph extractor. Empirically, we add CIA or CIA-LRA after one epoch of ERM training.

Algorithm 1 Detailed Training Procedure of CIA-LRA

Require:

A labeled training graph $\mathcal{G} = (A, X, Y)$, a GNN f_Θ, and invariant subgraph generator GNN f_{θ_m}. The number of hops t, CIA-LRA weight λ, the number of classes C, total iterations T, model learning rate r_1, invariant subgraph generator learning rate r_2, the number of f_Θ's layers L

Ensure:

Updated model f_Θ with parameter Θ.

1: **for** iterations in $[1, 2, ..., T]$ **do**
2:    Randomly sample a subgraph $A' \in \mathbb{R}^{N \times N}$ from $A \in \mathbb{R}^{N_0 \times N_0}$.
3:    Compute and apply the edge mask according to Equation (8) to obtain the masked adjacency matrix $A_m \leftarrow A' \odot M \in \mathbb{R}^{N \times N}$.
4:    Initialize $\mathcal{L}_{\text{CIA-LRA}} \leftarrow 0$.
5:    Calculate the node representations $\phi(A_m, X) \in \mathbb{R}^{N \times D}$.
6:    ### Calculate A^t, where the (i, j)-th element of A^t equals the length of the shortest path from node i to j if the length is less than t else infinity:
7:    Initialize $A^t(i, j) \leftarrow$ Inf if $i \neq j$, $A^t(i, i) \leftarrow 1$, $A_{tem} \leftarrow A_m$
8:    **for** hop h in $[1, 2, ..., t]$ **do**
9:       $A_{tem} \leftarrow A_m A_{tem}$
10:      **if** $A_{tem}(i, j) > 0$ **and** $A_{tem}(i, j) < A^t(i, j)$ **then**
11:        $A^t(i, j) \leftarrow h$
12:      **end if**
13:    **end for**
14:    ### Compute the ratio of neighbored nodes of each class:
15:    Compute the normalized adjacency matrix $\bar{A}$, where $\bar{A}$'s i-th row $\bar{A}_i \leftarrow A_{mi}/D_i$, A_{mi} is the i-th row of A_m and $D_i \in \mathbb{R}$ is the degree of node i.
16:    Initialize the neighbored label ratio $R \leftarrow Y \in \mathbb{R}^{N \times C}$, where $R(i, c)$ is the ratio of node i's neighbors of class c within a L-hop range, Y are the one hop labels.
17:    **for** l in $[1, 2, ..., L]$ **do**
18:       $R \leftarrow \bar{A}R$
19:    **end for**
20:    **for** c in $[1, 2, ..., C]$ **do**
21:      Sample the nodes of class c from A^t and form A_c^t. Use A_c^t to screen for pairs of nodes not exceeding a distance of t hops $\Omega_c(t)$.
22:      Compute CIA-LRA loss of class c: $\mathcal{L}_{\text{CIA-LRA}}^c$ according to Equation (9) using $\Omega_c(t)$, R, A_c^t and $\phi(A_m, X)$:
23:      $\mathcal{L}_{\text{CIA-LRA}} \leftarrow \mathcal{L}_{\text{CIA-LRA}} + \mathcal{L}_{\text{CIA-LRA}}^c$
24:    **end for**
25:    Compute final loss $\mathcal{L} \leftarrow \mathcal{L}_{\text{ce}}(f_\Theta(A, X), Y) + \lambda \mathcal{L}_{\text{CIA-LRA}}$, $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss
26:    Update model parameters $\Theta \leftarrow \Theta - r\nabla_\Theta \mathcal{L}$, $\theta_m \leftarrow \Theta_m - r\nabla_{\theta_m} \mathcal{L}$
27: **end for**

F Additional Discussion of Theoretical Settings and Results

F.1 Detailed Setup of the Theoretical Model in Section 2

The proposed data generation process. In the theoretical model of Equation (2), each dimension of $n_1 \in \mathbb{R}^{N^e \times 1}$ and $n_2 \in \mathbb{R}^{N^e \times 1}$ are i.i.d, following a standard Gaussian distribution. $\epsilon^e \in \mathbb{R}^{N^e \times 1}$ is an environment spurious variable. ϵ_i^e (each dimension of ϵ^e) are independent random variables, $i = 1, ..., N^e$. We further assume the cross-environment expectation $\mathbb{E}_e[\epsilon^e] = \mathbf{0}$ and cross-environment variance $\mathbb{E}_e[\epsilon_i^e] = \sigma^2$, $i = 1, ..., N^e$ for brevity.

30

The considered multi-layer GNN. In the analyzed GNN of Equation (3), we simplify the classifier to an identity mapping. Such simplification has been adopted by various previous theoretical works on graphs [Wu et al., 2022b, Tang and Liu, 2023]. We assume $L \geq k$ to ensure the model has enough capacity to learn invariant features. We verify this assumption by using GCNs with different numbers of layers to predict the ground-truth labels (see Appendix D.6).

F.2 Discussion of the Structural Feature Considered in the Theoretical Model and Justification for the Choice of the Analyzed GNN

Structural Features and Structural Shifts Considered in Section 2.1. To reflect reality as much as possible, it is necessary to consider both nodal and structural invariant and spurious features in the theoretical model. As mentioned in Section 2.1, we model the invariant structural feature as the structure of the k-hop ego-subgraph. A natural question is raised here:

Can we find other ways to define the invariant/spurious structural features?

The answer is yes. For example, the invariant structure can be modeled as the subgraph of the ego-graph of a node, following Li et al. [2023a]. However, it is fundamentally impossible for GNNs using mean aggregation (like GCN) to learn such causal structures. This is because such GNNs will assign fixed weights to each neighboring node feature, and they can't split the causal substructure from the neighbored ego-sgraph. Therefore, we make the causal structure feasible for GCN-like GNNs to learn by defining the causal structure as the whole k-hop neighboring ego-graph, rather than a subgraph, and show that OOD failure can still happen (Theorem 2.3). Then, under this setting, the remaining challenge becomes identifying the true k by optimizing the shallow layer GNN parameters. However, in real practice, the invariant causal pattern may still be an ego-subgraph. This can be reflected in the performance gain of the invariant subgraph extractor used in CIA-LRA.

Why Do We Choose Such a GNN in Section 2.1? From the above analysis, we show that such a choice is a compromise solution between the case of GCN (that can only extract a whole ego-graph) and GAT-like GNNs (that can extract a subgraph from an ego-graph). Although in this GNN each neighbored node is solely assigned the same weight, the shallow layer parameters can be optimized to realize the aggregation of different depths to capture the causal structures of different depths.

F.3 Discussion of the Failure Solution for GNNs of VREx and IRMv1

In Theorem G.2 and G.3 (the formal version Theorem 2.3), we show that VREx and IRMv1 could induce a model that uses spurious features. Now we'll give an intuitive explanation of this failure mode. When the lower-layer parameters of the GNN $\theta_1^{1\,(l)}, \theta_2^{1\,(l)}, \theta_1^{2\,(l)}, \theta_2^{2\,(l)}$ take the specific solution Θ_0 in Equation (21), we have

$$H_1^{(L)} = \frac{\partial H_1^{(L)}}{\partial \theta_1^{i\,(l)}} = \tilde{A}e^s X_1, \ H_2^{(L)} = \frac{\partial H_2^{(L)}}{\partial \theta_2^{i\,(l)}} = \tilde{A}e^{k+m} X_1, \tag{25}$$

holds for $i = 1, 2, \ l = 1, ..., L-1$ and every environment e. Thus, we get

$$\frac{\partial \mathcal{L}}{\partial \theta_1} = \frac{\partial \mathcal{L}}{\partial (H_1^{(L)}\theta_1)} \frac{\partial (H_1^{(L)}\theta_1)}{\partial \theta_1} = \frac{\partial \mathcal{L}}{\partial (H_1^{(L)}\theta_1)} H_1^{(L)}$$

$$\overset{(*)}{=} \frac{\partial \mathcal{L}}{\partial (H_1^{(L)}\theta_1)} \frac{\partial (H_1^{(L)})}{\partial \theta_1^{i\,(l)}} = \frac{\partial \mathcal{L}}{\partial \theta_1^{i\,(l)}} \frac{1}{\theta_1}, \tag{26}$$

$$i = 1, 2, \quad l = 1, ..., L-1$$

$(*)$ is because of Equation (25). Therefore, $\frac{\partial \mathcal{L}}{\partial \theta_1} = 0 \Rightarrow \frac{\partial \mathcal{L}}{\partial \theta_1^{i\,(l)}} = 0$. The same is true for $\frac{\partial \mathcal{L}}{\partial \theta_2}$ and $\frac{\partial \mathcal{L}}{\partial \theta_2^{i\,(l)}}$. This means the solution of the top-level parameters θ_1 and θ_2 of the GNN will only be constrained by two equations, $\frac{\partial \mathcal{L}}{\partial \theta_1} = 0$ and $\frac{\partial \mathcal{L}}{\partial \theta_2} = 0$, rather than be constrained by all gradient functions $\frac{\partial \mathcal{L}}{\partial \theta_i^j} = 0, \ i = 1, 2$. By analyzing the specific loss of VREx and IRMv1, we conclude that they will induce a non-zero θ_2.

31

Note that the failure solution Θ_0 here is not the unique one, we choose Θ_0 just for the elegant expression and to better convey the intuition. In effect, the conclusion $\frac{\partial \mathcal{L}}{\partial \theta_1} = 0 \Rightarrow \frac{\partial \mathcal{L}}{\partial \theta_1^{i}{}^{(l)}} = 0$ and $\frac{\partial \mathcal{L}}{\partial \theta_2} = 0 \Rightarrow \frac{\partial \mathcal{L}}{\partial \theta_2^{i}{}^{(l)}} = 0$ holds as long as the lower-layer aggregation parameters satisfy

$$\Theta_0' = \begin{cases} \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s_1+1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s_1, L-s_1-1, ..., 1 \\ \theta_2^{1(l)} = 1, \theta_2^{2(l)} = 1, & l = L-1, ..., L-s_2+1 \\ \theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-s_2, L-s_2-1, ..., 1 \end{cases}, \text{ for some } s_1,\ s_2 \in \mathbb{N}^+,\ 1 < s_1 \le L,\ 1 < s_2 \le L,$$

(27)

i.e., we don't require the spurious branch of the GNN to be identity mapping I as equation (36) does.

This failure mode can happen to GCN (all $\theta_j^i = 1$, $i = 1, 2$, $j = 1, 2$) and also for GAT.

F.4 The Superiority of the Proposed CSBM-OOD

Our CSBM-OOD in Section introduces several advancements over the conventional CSBMs [Ma et al., 2021, Mao et al., 2023]: 1) It supports multi-class classification, extending beyond the binary classification framework of traditional CSBMs; 2) It accommodates unique neighboring label distributions for each node, in contrast to the traditional models that assume a uniform class-shared homophily/heterophily ratio across all nodes; 3) our model integrates OOD shifts, while traditional ones don't.

F.5 Tightness of the Error Bound of Theorem 4.4

When there are no distributional shifts in spurious node features and heterophilic neighborhood distribution between training and test environments, the terms (a)-(d) in Eq. (109) becomes zero, and the upper bound becomes $\widehat{\mathcal{L}}_{e^{tr}}^{\gamma}(\tilde{h}) + const = \widehat{\mathcal{L}}_{e^{tr}}^{\gamma}(\tilde{h}) + \frac{1}{N_{e^{tr}}^{1-2\alpha}} + \frac{1}{N_{e^{tr}}^{2\alpha}} \ln \frac{LC(2B_{e^{te}})^{1/L}}{\gamma^{1/L}\delta}$, i.e., our bound only larger than the ideal error $\widehat{\mathcal{L}}_{e^{tr}}^{\gamma}(\tilde{h})$ by a constant $const$. When the number of training samples $N_{e^{tr}}$ is large, $const$ will be small enough and can be negligible. Hence, the tightness of our bound is guaranteed.

G Proofs of the Theoretical Results

G.1 Proofs of the Concept Shift Case Presented in the Main Text

In this section, we give proof of the propositions of the concept shift model presented in the main text.

Proposition G.1. *For the non-graph version of the SCM in Equation (2),*

$$Y^e = X_1 + n_1,\ X_2^e = Y^e + n_2 + \epsilon^e,$$

(28)

VREx and IRMv1 will learn invariant features when using a 1-layer linear network: $f(X) = \theta_1 X_1 + \theta_2 X_2$.

Proof. **VREx.** Denote $X_1\theta_1 + X_2^e\theta_2 - X_1 - n_1$ as l_e. The variance of loss across environments is:

$$\begin{aligned} \mathbb{V}_e[R(e)] &= \mathbb{E}_e[R^2(e)] - \mathbb{E}_e^2[R(e)] \\ &= \mathbb{E}_e\left[\left(\mathbb{E}_{n_1,n_2}\|X_1\theta_1 + (X_1 + n_1 + n_2 + \epsilon^e)\theta_2 - X_1 - n_1\|_2^2\right)^2\right] \\ &\quad - \mathbb{E}_e^2\left[\mathbb{E}_{n_1,n_2}\|X_1\theta_1 + (X_1 + n_1 + n_2 + \epsilon^e)\theta_2 - (X_1 - n_1\|_2^2\right]. \\ &= \mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[(l_e^\top l_e)^2\right]\right] - \mathbb{E}_e^2\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]. \end{aligned}$$

(29)

32

Take the derivative of $\mathbb{V}_e[R(e)]$ with respect to θ_1:

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = \mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\mathbb{E}_{n_1,n_2}\left[2l_e^\top X_1\right]\right]$$
$$- 2\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[2l_e^\top X_1\right]\right] \tag{30}$$

Using the fact that $\mathbb{E}_{n_1,n_2}[n_1] = \mathbb{E}_{n_1,n_2}[n_2] = \mathbf{0}$, $\mathbb{E}_{n_1,n_2}[n_1^\top n_2] = \mathbb{E}_{n_1,n_2}[n_2^\top n_1] = 0$ and $\mathbb{E}_{n_1,n_2}[n_1^\top n_1] = \mathbb{E}_{n_1,n_2}[n_2^\top n_2] = N^e$ if it is the noise from e, $\mathbb{E}_{n_1,n_2}[n_1^\top \epsilon^e] = \mathbb{E}_{n_1,n_2}[n_2^\top \epsilon^e] = 0$ and using the assumption that $\mathbb{E}_e[(\epsilon_i^e)^2] = \sigma^2$, we have

$$\mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\mathbb{E}_{n_1,n_2}\left[2l_e^\top X_1\right]\right]$$
$$= \theta_1{}^3 X_1{}^4 + 3\theta_1{}^2\theta_2 X_1{}^4 + \theta_1\theta_2{}^2[3X_1^2(X_1^2+\sigma^2) + 2X_1^2\mathbb{E}_e[N^e]]$$
$$+\theta_2^3 X_1(X_1^3 + X_1\sigma^2 + \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^e] + 2X_1\mathbb{E}_e[N^e])$$
$$-3\theta_1^2 X_1^4 - \theta_1\theta_2(6X_1^4 + 2X_1^2\mathbb{E}_e[N^e]) - \theta_2^2(3X_1^2(X_1^2+\sigma^2) + 4X_1^2\mathbb{E}_e[N^e]) \tag{31}$$
$$+\theta_1(3X_1^4 + X_1^2\mathbb{E}_e[N^e]0) + 3\theta_2(X_1^4 + X_1^2\mathbb{E}_e[N^e]) - X_1^2(X_1^2 + \mathbb{E}_e[N^e])$$

and

$$\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[2l_e^\top X_1\right]\right]$$
$$= \theta_1{}^3 X_1{}^4 + 3\theta_1{}^2\theta_2 X_1{}^4 + \theta_1\theta_2{}^2[3X_1^2(X_1^2+\sigma^2) + 2X_1^2\mathbb{E}_e[N^e]]$$
$$+\theta_2^3 X_1(X_1^3 + X_1\sigma^2 + 2X_1\mathbb{E}_e[N^e])$$
$$-3\theta_1^2 X_1^4 - \theta_1\theta_2(6X_1^4 + 2X_1^2\mathbb{E}_e[N^e]) - \theta_2^2(3X_1^2(X_1^2+\sigma^2) + 4X_1^2\mathbb{E}_e[N^e]) \tag{32}$$
$$+\theta_1(3X_1^4 + X_1^2\mathbb{E}_e[N^e]0) + 3\theta_2(X_1^4 + X_1^2\mathbb{E}_e[N^e]) - X_1^2(X_1^2 + \mathbb{E}_e[N^e])$$

Plug Equation (31) and (32) back into Equation (30), we have

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = \theta_2^3 X_1^\top \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^e] \tag{33}$$

Let $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = 0$, we have $\theta_2 = 0$.

Now we need to validate $\theta_2 = 0$ is also a solution to $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2} = 0$. Let's calculate $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2}$ when $\theta_2 = 0$:

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2} = \mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\mathbb{E}_{n_1,n_2}\left[2l_e^\top(X_1 + n_1 + n_2 + \epsilon^e)\right]\right]$$
$$- 2\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[2l_e^\top(X_1 + n_1 + n_2 + \epsilon^e)\right]\right]$$
$$= (\theta_1^2 X_1^2 - 2\theta_1 X_1^2 + X_1^2 + \mathbb{E}_e N^e)(\theta_1 X_1^2 - X_1^2 - \mathbb{E}_e N^e) \tag{34}$$
$$-(\theta_1^2 X_1^2 - 2\theta_1 X_1^2 + X_1^2 + \mathbb{E}_e N^e)(\theta_1 X_1^2 - X_1^2 - \mathbb{E}_e N^e)$$
$$= 0.$$

So far, we have proved $\theta_2 = 0$ is the solution for VREx, hence it will learn invariant features. We finish the proof for VREx.

IRMv1. The objective of IRMv1 is $\mathbb{E}_e\|\nabla_w R(e)\|_2^2$. When IRMv1 loss is optimized to zero, we have $\nabla_w R(e) = 0$ for all environments e.

$$\nabla_w R(e) = \mathbb{E}_{n_1,n_2}[2(\theta_1 X_1 + \theta_2 X_2 - (X_1 + n_1))^\top(\theta_1 X_1 + \theta_2 X_2)]$$
$$= 2((\theta_1)^2 X_1^\top X_1 + (\theta_2)^2(X_1^\top X_1 + \epsilon^{e\top}\epsilon^e + 2N^e) + 2\theta_1\theta_2 X_1^\top(X_1 + \epsilon^e) \tag{35}$$
$$- \theta_1 X_1^\top X_1 - \theta_2(X_1^\top X_1 + X_1^\top \epsilon^e + N^e))$$

To realize $\nabla_w R(e) = 0$ for all e, we must let $\theta_2 = 0$ (and consequently $\theta_1 = 1$), otherwise the solution of θ_2 will include terms related to ϵ^e and N^e that vary with environments, and a single value θ_2 cannot fit all these values. Thus we finish the proof for IRMv1. $\qquad\square$

33

G.1.2 Proof of the Failure Case on Graphs of VREx under Concept Shift

We present the formal version of the VREx part in Theorem 2.3 as Theorem G.2 below.

Theorem G.2. *(VREx will use spurious features on graphs under concept shift, formal) Under the SCM of Equation (2), the objective $\min_{\Theta} \mathbb{V}_e[R(e)]$ has non-unique solutions for parameters of the GNN (3) when part of the model parameters $\{\theta_1^{1^{(l)}}, \theta_1^{2^{(l)}}, \theta_2^{1^{(l)}}, \theta_2^{2^{(l)}}\}$ take the values*

$$
\Theta_0 = \begin{cases}
\theta_1^{1^{(l)}} = 1, \theta_1^{2^{(l)}} = 1, & l = L-1, ..., L-s+1 \\
\theta_1^{1^{(l)}} = 0, \theta_1^{2^{(l)}} = 1, & l = L-s, L-s-1, ..., 1 \\
\theta_2^{1^{(l)}} = 0, \theta_2^{2^{(l)}} = 1, & l = L-1, ..., 1
\end{cases}, \tag{36}
$$

for some $0 < s < L$. Specifically, the VREx solutions of θ_1 and θ_2 are the sets of solutions of the cubic equation, some of which are spurious solutions that $\theta_2 \neq 0$ (although $\theta_2 = 0$ is indeed one of the solutions, VREx is not guaranteed to reach this solution):

$$
\begin{cases}
(3c_1\theta_1\theta_2 + c_1(\theta_2)^2 - 2c_6\theta_2)\sigma^2 - \mathbb{E}_e[N^e(2c_1(\theta_1+\theta_2)-c_6)]\sigma^2\theta_2 + c_7\theta_2 = 0 \\
(\mathbb{E}_e[N^e(2c_1(\theta_1+\theta_2)-c_6)]\sigma^2\theta_2 - c_7)(c_3-c_4)\theta_2 - [c_2(\theta_1+\theta_2)-c_5](\theta_2)^2 = 0
\end{cases}. \tag{37}
$$

where $c_1 = \mathbb{E}_e[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)]$, $c_2 = \mathbb{E}_e[N^e(\tilde{A}^{e^s}X_1)^\top\tilde{A}^{e^s}X_1]$, $c_3 = \mathbb{E}_e[(\tilde{A}^{e^s}X_1)^\top\mathbf{1}]$, $c_4 = \mathbb{E}_e[((\tilde{A}^{e^k}X_1)^\top\mathbf{1}]$, $c_5 = \mathbb{E}_e[N^e((\tilde{A}^{e^k}X_1)^\top\tilde{A}^{e^s}X_1 + tr((\tilde{A}^{e^k})^\top\tilde{A}^{e^k}) + N^e(1 + \sigma^2))]$, $c_6 = \mathbb{E}_e[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^k}X_1)]$, $c_7 = \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]$.

Proof. We will use some symbols to simplify the expression of the toy GNN. Denote $\tilde{A}^m n_1 + n_2 + \epsilon$ as η. Use the following notations to represent the components of the L-layer GNN model:

$$
\begin{aligned}
f_\Theta(A, X) &= H_1^{(L)}\theta_1 + H_2^{(L)}\theta_2 \\
&= \underbrace{\left[\theta_1^{1^{(L-1)}}\bar{A}\left(...\theta_1^{1^{(3)}}\left(\theta_1^{1^{(2)}}\bar{A}(\theta_1^{1^{(1)}}\bar{A} + \theta_1^{2^{(1)}}\bar{I})X_1 + \theta_1^{2^{(2)}}(\theta_1^{1^{(1)}}\bar{A} + \theta_1^{2^{(1)}}\bar{I})X_1\right) + ...\right)\right]}_{C_1}\theta_1 \\
&\quad + \underbrace{\left[\theta_2^{1^{(L-1)}}\bar{A}\left(...\theta_2^{1^{(3)}}\left(\theta_2^{1^{(2)}}\bar{A}(\theta_2^{1^{(1)}}\bar{A} + \theta_2^{2^{(1)}}\bar{I})\tilde{A}^{k+m}X_1 + \theta_2^{2^{(2)}}(\theta_2^{1^{(1)}}\bar{A} + \theta_2^{2^{(1)}}\bar{I})\tilde{A}^{k+m}X_1\right) + ...\right)\right]}_{C_2}\theta_2 \\
&\quad + \underbrace{\left[\theta_2^{1^{(L-1)}}\bar{A}\left(...\theta_2^{1^{(3)}}\left(\theta_2^{1^{(2)}}\bar{A}(\theta_2^{1^{(1)}}\bar{A} + \theta_2^{2^{(1)}}\bar{I})\eta + \theta_2^{2^{(2)}}(\theta_2^{1^{(1)}}\bar{A} + \theta_2^{2^{(1)}}\bar{I})\eta\right) + ...\right)\right]}_{Z}\theta_2 \\
&= C_1\theta_1 + (C_2 + Z)\theta_2.
\end{aligned}
\tag{38}
$$

$C_1, C_2, Z \in \mathbb{R}^{N \times 1}$. We use C_1^e, C_2^e, and Z^e to denote the variables from the corresponding environment e. We further denote $C_2^e = C_2^{e'}\tilde{A}^{e^s}X_1$, $Z^e = C_2^{e'}\eta$.

Using these notations, the loss of environment e is

$$
\begin{aligned}
R(e) &= \mathbb{E}_{n_1,n_2}\left[\|f_\Theta(A^e, X^e) - Y^e\|_2^2\right] \\
&= \mathbb{E}_{n_1,n_2}\left[\left\|C_1^e\theta_1 + (C_2^e + Z^e)\theta_2 - \tilde{A}^{e^k}X_1 - n_1\right\|_2^2\right].
\end{aligned}
\tag{39}
$$

Denote the inner term $C_1^e\theta_1 + (C_2^e + Z^e)\theta_2 - \tilde{A}^{e^k}X_1 - n_1$ as l_e.

The variance of loss across environments is:

$$
\begin{aligned}
\mathbb{V}_e[R(e)] &= \mathbb{E}_e[R^2(e)] - \mathbb{E}_e^2[R(e)] \\
&= \mathbb{E}_e\left[\left(\mathbb{E}_{n_1,n_2}\left\|C_1^e\theta_1 + (C_2^e + Z^e)\theta_2 - \tilde{A}^{e^k}X_1 - n_1\right\|_2^2\right)^2\right] \\
&\quad - \mathbb{E}_e^2\left[\mathbb{E}_{n_1,n_2}\left\|C_1^e\theta_1 + (C_2^e + Z^e)\theta_2 - \tilde{A}^{e^k}X_1 - n_1\right\|_2^2\right]. \\
&= \mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[(l_e^\top l_e)^2\right]\right] - \mathbb{E}_e^2\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right].
\end{aligned}
\tag{40}
$$

34

Take the derivative of $\mathbb{V}_e[R(e)]$ with respect to θ_1:

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = \mathbb{E}_e \left[2\mathbb{E}_{n_1,n_2} \left[l_e^\top l_e \right] \mathbb{E}_{n_1,n_2} \left[2l_e^\top C_1^e \right] \right]$$
$$- 2\mathbb{E}_e \left[\mathbb{E}_{n_1,n_2} \left[l_e^\top l_e \right] \right] \mathbb{E}_e \left[\mathbb{E}_{n_1,n_2} \left[2l_e^\top C_1^e \right] \right] \tag{41}$$

Calculate the derivative by terms:

$$\mathbb{E}_{n_1,n_2}[l_e^\top l_e] = \mathbb{E}_{n_1,n_2}[C_1^{e\top} C_1^e (\theta_1)^2 + C_1^{e\top} C_2^e \theta_1 \theta_2 + C_1^{e\top} Z^e \theta_1 \theta_2 - C_1^{e\top} (\tilde{A}e)^k X_1 \theta_1 - C_1^{e\top} n_1 \theta_1$$
$$+ C_2^{e\top} C_1^e \theta_1 \theta_2 + C_2^{e\top} C_2^e (\theta_2)^2 + C_2^{e\top} Z^e \theta_1 \theta_2 - C_2^{e\top} (\tilde{A}e)^k X_1 \theta_2 - C_2^{e\top} n_1 \theta_2$$
$$+ Z^{e\top} C_1^e \theta_1 \theta_2 + Z^{e\top} C_2^e (\theta_2)^2 + Z^{e\top} Z^e (\theta_2)^2 - Z^{e\top} (\tilde{A}e)^k X_1 \theta_2 - Z^{e\top} n_1 \theta_2$$
$$- ((\tilde{A}e)^k X_1)^\top (C_1^e \theta_1 + C_2^e \theta_2) - ((\tilde{A}e)^k X_1)^\top Z^e \theta_2 + ((\tilde{A}e)^k X_1)^\top (\tilde{A}e)^k X_1$$
$$+ ((\tilde{A}e)^k X_1)^\top n_1 - n_1^\top (C_1^e \theta_1 + C_2^e \theta_2) - n_1^\top Z^e \theta_2 + n_1^\top (\tilde{A}e)^k X_1 + n_1^\top n_1] \tag{42}$$

Since n_1 and n_2 are independent standard Gaussian noise, we have $\mathbb{E}_{n_1,n_2}[n_1] = \mathbb{E}_{n_1,n_2}[n_2] = \mathbf{0}$, $\mathbb{E}_{n_1,n_2}[n_1^\top n_2] = \mathbb{E}_{n_1,n_2}[n_2^\top n_1] = 0$ and $\mathbb{E}_{n_1,n_2}[n_1^\top n_1] = \mathbb{E}_{n_1,n_2}[n_2^\top n_2] = N^e$ if it is the noise from e. Also, since ϵ^e and n_1, n_2 are independent, we have $\mathbb{E}_{n_1,n_2}[n_1^\top \epsilon^e] = \mathbb{E}_{n_1,n_2}[n_2^\top \epsilon^e] = 0$.

When

$$\begin{cases} \theta_1^{1\,(l)} = 1, \theta_1^{2\,(l)} = 1, & l = L-1, ..., L-s+1 \\ \theta_1^{1\,(l)} = 0, \theta_1^{2\,(l)} = 1, & l = L-s, L-s-1, ..., 1 \\ \theta_2^{1\,(l)} = 0, \theta_2^{2\,(l)} = 1, & l = L-1, ..., 1 \end{cases} \tag{43}$$

we have $C_2^{e\prime} = I_{N^e} \in \mathbb{R}^{N^e \times N^e}$ and $C_1^e = \tilde{A}e^s X_1$. Consequently, we get $\mathbb{E}_{n_1,n_2}[Z^{e\top} n_1] = \mathrm{tr}(C_2^{e\prime} \tilde{A}e^k) = \mathrm{tr}(\tilde{A}e^k)$, $\mathbb{E}_{n_1,n_2}[Z^{e\top} Z^e] = \mathrm{tr}\left((\tilde{A}e^k)^\top (\tilde{A}e^k) \right) + N^e + \epsilon^{e\top} \epsilon^e$.

Use the above conclusions and rewrite Equation (42) as (here we only plug in the value of $C_2^{e\prime}$):

$$\mathbb{E}_{n_1,n_2}[l_e^\top l_e] =$$
$$\left. \begin{array}{l} C_1^{e\top} C_1^e (\theta_1)^2 + C_1^{e\top} C_2^e \theta_1 \theta_2 - C_1^{e\top} (\tilde{A}e)^k X_1 \theta_1 + C_2^{e\top} C_1^e \theta_1 \theta_2 + C_2^{e\top} C_2^e (\theta_2)^2 - C_2^{e\top} (\tilde{A}e)^k X_1 \theta_2 \\ +\mathrm{tr}\left((\tilde{A}e^k)^\top (\tilde{A}e^k) \right) (\theta_2)^2 - ((\tilde{A}e)^k X_1)^\top (C_1^e \theta_1 + C_2^e \theta_2) + ((\tilde{A}e)^k X_1)^\top (\tilde{A}e)^k X_1 + N^e \left(1 + (\theta_2)^2 \right) \\ -2\mathrm{tr}(\tilde{A}e^k) \end{array} \right\} (*)$$
$$+ [C_1^{e\top} \epsilon^e + C_2^{e\top} \epsilon^e + \epsilon^{e\top} C_1^e] \theta_1 \theta_2 + \epsilon^{e\top} \epsilon^e (\theta_2)^2 - 2((\tilde{A}e)^k X_1)^\top \epsilon^e \theta_2 \} (**), \tag{44}$$

$(*)$ and $(**)$ represent terms that are independent and associated with ϵ^e, respectively. Additionally,

$$\mathbb{E}_{n_1,n_2}[2l_e^\top C_1^e] = 2 \left[C_1^{e\top} C_1^e \theta_1 + C_2^{e\top} C_1^e \theta_2 + (C_2^{e\prime} \epsilon^e)^\top C_1^e \theta_2 - ((\tilde{A}e)^k X_1)^\top C_1^e \right] \tag{45}$$
$$= 2 \left[C_1^{e\top} C_1^e \theta_1 + C_2^{e\top} C_1^e \theta_2 + \epsilon^{e\top} C_1^e \theta_2 - ((\tilde{A}e)^k X_1)^\top C_1^e \right].$$

Multiplying Equation (44) and (45) and take the expectation on e, using the assumption that $\mathbb{E}_e[(\epsilon^e{}_i)^2] = \sigma^2$ ($\epsilon^e{}_i$ is the i-th element of ϵ^e):

$$\mathbb{E}_e \left[2\mathbb{E}_{n_1,n_2}[l_e^\top l_e] \mathbb{E}_{n_1,n_2} \left[2l_e^\top C_1 \right] \right] = 4\mathbb{E}_e \left[(*) \left(C_1^{e\top} C_1^e \theta_1 + C_2^{e\top} C_2^{e\prime} \theta_2 - ((\tilde{A}e)^k X_1)^\top C_1^e \right) \right]$$
$$+ 4\mathbb{E}_e \left[(\tilde{A}e^s X_1)^\top \tilde{A}e^s X_1 (3\theta_1 \theta_2 + (\theta_2)^2) - 2(\tilde{A}e^s X_1)^\top ((\tilde{A}e)^k X_1) \theta_2 \right] \theta_2 \sigma^2$$
$$+ 4\mathbb{E}_e[N^e \epsilon^{e\top} \epsilon^e \epsilon^e{}^\top (\tilde{A}e^s X_1)] \theta_2. \tag{46}$$

Next target is to compute $2\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[l_e^\top l_e]]$ and $\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[2l_e^\top C_1]]$. Since ϵ^e has zero mean, we have:

$$2\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[l_e^\top l_e]] = 2\mathbb{E}_e[(*)] + 2\mathbb{E}_e[N^e](\theta_2)^2 \sigma^2 \tag{47}$$

35

and

$$\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[2l_e^\top C_1^e]] = 2\mathbb{E}_e\left[C_1^{e\top}C_1^e\theta_1 + C_2^{e\top}C_1^e\theta_2 - ((\tilde{A}^e)^k X_1)^\top C_1^e\right]. \tag{48}$$

Use Equation (46) (47) and (48) and let $\frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_1} = 0$, we have:

$$\mathbb{E}_e\left[3(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)(\theta_1\theta_2 + \frac{1}{3}(\theta_2)^2) - 2(\tilde{A}^{e^s}X_1)^\top((\tilde{A}^e)^k X_1)\theta_2\right]\sigma^2 + \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\theta_2$$

$$-\mathbb{E}_e[N^e]\mathbb{E}_e\left[2(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)(\theta_1 + \theta_2) - ((\tilde{A}^e)^k X_1)^\top C_1^e\right]\theta_2\sigma^2 = 0. \tag{49}$$

Now we start calculating the expression of $\frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_2}$:

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_2} = \mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\mathbb{E}_{n_1,n_2}\left[2l_e^\top(C_2 + Z^e)\right]\right] \tag{50}$$
$$- 2\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[2l_e^\top(C_2^e + Z^e)\right]\right].$$

Let $\frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_2} = 0$:

$$\mathbb{E}_e\left[(C_1^{e\top}C_2^{e\prime} + C_2^{e\top}C_2^{e\prime} + C_2^{e\prime\top}C_1^{e\top})\theta_1\theta_2 + (C_2^{e\prime})^\top C_2^{e\prime}(\theta_2)^2 - 2((\tilde{A}^e)^k X_1)^\top C_2^{e\prime}\theta_2\right]$$

$$\mathbb{E}_e\left[(C_2^{e\prime\top}C_2^e\theta_2 - ((\tilde{A}^e)^k X_1)^\top C_2^{e\prime})\right]\sigma^2$$

$$- \mathbb{E}_e\left[N^e\sigma^2\left(C_1^{e\top}C_2^e\theta_1 + C_2^{e\top}C_2^e\theta_2 - ((\tilde{A}^e)^k X_1)^\top C_2^e + \mathrm{tr}((\tilde{A}^{e^k})^\top \tilde{A}^{e^k}) + N^e + C_2^{e\prime\top}C_2^{e\prime}\sigma^2\right)(\theta_2)^2\right]$$

$$= 0. \tag{51}$$

Plug Equation (49) in (51), we reach:

$$\left[\mathbb{E}_e\left[N^e(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)(\theta_1 + \theta_2) - ((\tilde{A}^e)^k X_1)^\top C_1^e\right]\sigma^2 - \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\right]\theta_2$$

$$\mathbb{E}_e\left((\tilde{A}^{e^s}X_1)^\top \mathbf{1}_{\mathbf{N}^e}\theta_2 - (\tilde{A}^{e^k}X_1)^\top \mathbf{1}_{\mathbf{N}^e}\right)$$

$$- \mathbb{E}_e\left[N^e\left((\tilde{A}^{e^s}X_1)^\top \tilde{A}^{e^s}X_1(\theta_1 + \theta_2) - (\tilde{A}^{e^k}X_1)^\top \tilde{A}^{e^s}X_1 + \mathrm{tr}((\tilde{A}^{e^k})^\top \tilde{A}^{e^k}) + N^e(1 + \sigma^2)\right)\right](\theta_2)^2$$

$$= 0. \tag{52}$$

Let $c_1 = \mathbb{E}_e[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)]$, $c_2 = \mathbb{E}_e[N^e(\tilde{A}^{e^s}X_1)^\top\tilde{A}^{e^s}X_1]$, $c_3 = \mathbb{E}_e[(\tilde{A}^{e^s}X_1)^\top\mathbf{1}]$, $c_4 = \mathbb{E}_e[((\tilde{A}^{e^k}X_1)^\top\mathbf{1}]$, $c_5 = \mathbb{E}_e[N^e((\tilde{A}^{e^k}X_1)^\top\tilde{A}^{e^s}X_1 + \mathrm{tr}((\tilde{A}^{e^k})^\top\tilde{A}^{e^k}) + N^e(1 + \sigma^2))]$, $c_6 = \mathbb{E}_e[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^k}X_1)]$, $c_7 = \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]$,

we conclude that

$$\begin{cases} (3c_1\theta_1\theta_2 + c_1(\theta_2)^2 - 2c_6\theta_2)\sigma^2 - \mathbb{E}_e[N^e(2c_1(\theta_1 + \theta_2) - c_6)]\sigma^2\theta_2 + c_7\theta_2 = 0 \\ (\mathbb{E}_e[N^e(2c_1(\theta_1 + \theta_2) - c_6)]\sigma^2\theta_2 - c_7)(c_3 - c_4)\theta_2 - [c_2(\theta_1 + \theta_2) - c_5](\theta_2)^2 = 0 \end{cases}. \tag{53}$$

As for the derivative respect to $\theta_1^{1\,(l)}, \theta_1^{2\,(l)}, \theta_2^{1\,(l)}, \theta_2^{2\,(l)}$, when they take the special value in Equation (43), we have $\frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_1} = 0 \Rightarrow \frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_1^{1\,(l)}} = \frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_1^{2\,(l)}} = 0$ and $\frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_2} = 0 \Rightarrow \frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_2^{1\,(l)}} = \frac{\partial \mathbb{V}_e[R(e)]}{\partial\theta_2^{2\,(l)}} = 0, l = 1, ..., L$. So we conclude the solution induced by Equation (53) is the solution of the objective, and $\theta_2 = 0$ is not a valid solution.

$\square$

36

G.1.3 Proof of the Failure Case on Graphs of IRMv1 under Concept Shift

We present the formal version of the IRM part in Theorem 2.3 as Theorem G.3 below.

Theorem G.3. *(IRMv1 will use spurious features on graphs under concept shift, formal) Under the SCM of Equation (2), there exists $s \in \mathbb{N}^+$ that satisfies $0 < s < L$ and $s \neq k$ such that optimizing the IRMv1 objective $\min_\Theta \mathbb{E}_e[\|\nabla_{w|w=1.0} R(e)\|^2]$ will not lead to the invariant solution $\theta_2 = 0$ for parameters of the GNN (3) when $\{\theta_1^{1(l)}, \theta_1^{2(l)}, \theta_2^{1(l)}, \theta_2^{2(l)}\}$ take the special solution:*

$$\Theta_0 = \begin{cases} \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s+1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s, L-s-1, ..., 1 \\ \theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-1, ..., 1 \end{cases}. \tag{54}$$

Proof. From the proof of non-graph IRMv1 case Appendix G.1.1 we know that when IRMv1 objective is optimized, we have $\nabla_w R(e) = 0$ for all e. For the graph case, the expected risk of environment e is

$$R(e) = \mathbb{E}_{n_1, n_2}[\|\theta_1 C_1^e + \theta_2(C_2^e + Z^e) - (\tilde{A}^e)^k X_1 - n_1\|_2^2], \tag{55}$$

where the definition of C_1^e, C_2^e and Z^e follows Equation (38). Now let's check if the invariant solution $\theta_2 = 0$ is a valid solution. If $\theta_2 = 0$ holds, then the following equation must hold for every environment e:

$$\begin{aligned}
\nabla_w R(e) &= \mathbb{E}_{n_1, n_2}[(\theta_1 C_1^e + \theta_2(C_2^e + Z^e) - (\tilde{A}^e)^k X_1 - n_1)^\top (\theta_1 C_1^e + \theta_2(C_2^e + Z^e))] \\
&= \mathbb{E}_{n_1, n_2}[(\theta_1)^2 C_1^{e\top} C_1^e + (\theta_2)^2 (C_2^e + Z^e)^\top (C_2^e + Z^e) + 2\theta_1\theta_2 C_1^{e\top}(C_2^e + Z^e) \\
&\quad - \theta_1 C_1^{e\top}((\tilde{A}^e)^k X_1 + n_1) - \theta_2(C_2^e + Z^e)((\tilde{A}^e)^k X_1 + n_1)] \\
&= (\theta_1)^2 ((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^s X_1) - \theta_1((\tilde{A}^e)^k X_1)^\top ((\tilde{A}^e)^s X_1)
\end{aligned} \tag{56}$$

When $s \neq k$, we have $\theta_1 = \frac{((\tilde{A}^e)^k X_1)^\top ((\tilde{A}^e)^s X_1)}{((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^s X_1)}$. The value of this solution of θ_1 varies with environment e, and thus is not a valid solution.

However, now we will show that optimizing IRMv1 does not necessarily lead to lower-layer parameters such that $s = k$. To reveal this, by taking the derivative of $\mathcal{L}_{\text{IRMv1}}$ w.r.t. θ_1 and θ_2 and let them $= 0$, we can get two cubic equations:

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1} =& \mathbb{E}_e[(C_1^{e\top} C_1^e(\theta_1)^2 + (C_2^{e\top} C_2^e + 2C_2^{e\top}\epsilon^e + 2N^e + \epsilon^\top\epsilon)(\theta_2)^2 \\
&+ (C_1^{e\top} C_2^e + C_1^{e\top}\epsilon^e)\theta_1\theta_2 - (\tilde{A}^{e^k} X_1)^\top C_1^e\theta_1 - [(\tilde{A}^{e^k} X_1)^\top + N^e + n_1^\top\epsilon^e]\theta_2) \\
&(2C_1^{e\top} C_1^{e\top}\theta_1 + (C_1^{e\top} C_2^e + C_1^{e\top}\epsilon)\theta_2 - (\tilde{A}^{e^k} X_1)^\top C_1^e)] = 0
\end{aligned} \tag{57}$$

and

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2} =& \mathbb{E}_e[(C_1^{e\top} C_1^e(\theta_1)^2 + (C_2^{e\top} C_2^e + 2C_2^{e\top}\epsilon^e + 2N^e + \epsilon^\top\epsilon)(\theta_2)^2 \\
&+ (C_1^{e\top} C_2^e + C_1^{e\top}\epsilon^e)\theta_1\theta_2 - (\tilde{A}^{e^k} X_1)^\top C_1^e\theta_1 - [(\tilde{A}^{e^k} X_1)^\top + N^e + n_1^\top\epsilon^e]\theta_2) \\
&(2(C_2^{e\top} C_2^e + 2C_2^{e\top}\epsilon^e + 2N^e + \epsilon^{e\top}\epsilon^e)\theta_2 + (C_1^{e\top} C_2^e + C_1^{e\top}\epsilon)\theta_1 \\
&- (\tilde{A}^{e^k} X_1)^\top (C_2^e + \epsilon^e) + N^e + n_1^\top\epsilon^e)] = 0
\end{aligned} \tag{58}$$

From the analysis in Appendix F.3, we know that as long as the lower-layer parameters take any value that satisfies the form in Equation (27), even if $s \neq k$, we can get $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^2}$ and $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^2}$. Thus, IRM cannot necessarily learn a $s = k$. At this time (when $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^2} = 0$ and $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^2} = 0$ but $s \neq k$), from the form of Equations (57) and (58) we know that there exist solutions that $\theta_2 \neq 0$, and the solution of θ_1 and θ_2 both depend on $\mathbb{E}_e(F(e))$, where $F(e)$ is some random variable associated with e.

$\square$

37

G.1.4 Proof of the Successful on Graphs Case of CIA under Concept Shift

Theorem G.4. *Optimizing the CIA objective will lead to the optimal solution Θ^*:*

$$
\begin{cases}
\theta_1 = 1 \\
\theta_2 = 0 \quad and \quad \exists l \in \{1, ..., L-1\} \ s.t. \ \theta_2^{1(l)} = \theta_2^{2(l)} = 0 \\
\theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, \quad l = L-1, ..., L-k+1 \\
\theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, \quad l = L-k, L-k-1, ..., 1
\end{cases}
\tag{59}
$$

Proof. For brevity, denote a node representation of $C_{1_c}^e$ as C_1^i and the one of $C_{1_c}^{e'}$ as C_1^j. The same is true for C_2^i and C_2^j. In this toy model, we need to consider the expectation of the noise, while in real cases such noise is included in the node features so taking expectation on e will handle this. Therefore, we add $\mathbb{E}_{n_1, n_2}$ in this proof, and this expectation is excluded in the formal description of the objective in the main paper.

$$
\begin{aligned}
\mathcal{L}_{\text{CIA}} &= \mathbb{E}_{\substack{e,e' \\ e \neq e'}} \mathbb{E}_{n_1,n_2} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega^{e,e'}}} \left[\mathcal{D}(\phi_\Theta(A^e, X^e)_{[c][v_i]}, \phi_\Theta(A^e, X^{e'})_{[c][v_j]}) \right] \\
&= \mathbb{E}_{\substack{e,e' \\ e \neq e'}} \mathbb{E}_{n_1,n_2} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega^{e,e'}}} \| C_1^i \theta_1 + (C_2^i + Z^e)\theta_2 - C_1^j \theta_1 - (C_2^j + Z^{e'})\theta_2 \|_2^2
\end{aligned}
\tag{60}
$$

$$
\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1} = \mathbb{E}_{\substack{e,e' \\ e \neq e'}} \mathbb{E}_{n_1,n_2} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega^{e,e'}}} \left[C_1^i \theta_1 + (C_2^i + Z^e)\theta_2 - C_1^j \theta_1 - (C_2^j + Z^{e'})\theta_2 \right]^\top (C_1^i - C_1^j)
\tag{61}
$$

Let $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1} = 0$, we have:

$$
\mathbb{E}_{\substack{e,e' \\ e \neq e'}} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega^{e,e'}}} \left[(C_1^i - C_1^j)^\top (C_1^i - C_1^j)\theta_1 + (C_2^i - C_2^k)^\top (C_1^i - C_1^j)\theta_2 \right] = 0
\tag{62}
$$

Also, we have:

$$
\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2} = \mathbb{E}_{\substack{e,e' \\ e \neq e'}} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega^{e,e'}}} \left[(C_1^i - C_1^j)^\top (C_2^i - C_2^j)\theta_1 + \left[(C_2^i - C_2^k)^\top (C_2^i - C_2^j) + (Z^e - Z^{e'})^\top (Z^e - Z^{e'}) \right] \theta_2 \right]
\tag{63}
$$

Further let $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2} = 0$, combining Equation (62) and using Assumption 2.1 we get

$$
\begin{cases}
\theta_1 = 0 \quad \text{or} \quad \exists l \in \{1, ..., L-1\} \ s.t. \ \theta_1^{1(l)} = \theta_1^{2(l)} = 0 \\
\theta_2 = 0
\end{cases}
\tag{64}
$$

or, if $\theta_1 \neq 0$ and $\forall l \in \{1, ..., L-1\}$, the parameters of that layer l of the invariant branch of the GNN are not all zero: $\theta_1^{1(l)} \neq 0$ or $\theta_1^{2(l)} \neq 0$, then we get

$$
\underbrace{\theta_2 \mathbb{E}_{\substack{e,e' \\ e \neq e'}} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j) \in \Omega^{e,e'}}} \left[-\frac{[(C_1^i - C_1^j)^\top (C_2^i - C_2^j)]^2}{(C_1^i - C_1^j)^\top (C_1^i - C_1^j)} + (C_2^i - C_2^j)^\top (C_2^i - C_2^j) + (Z^e - Z^{e'})^\top (Z^e - Z^{e'}) \right]}_{F} = 0
$$

$$
\tag{65}
$$

According to Cauchy–Schwarz inequality, $F > 0$ unless $\exists l \in \{1, ..., L-1\}$ s.t. $\theta_2^{1(l)} = \theta_2^{2(l)} = 0$. To ensure $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2}$, we conclude that $\theta_2 = 0$ or $\exists l \in \{1, ..., L-1\}$ s.t. $\theta_2^{1(l)} = \theta_2^{2(l)} = 0$.

In conclusion, to satisfy the constraint of CIA, no matter whether the invariant branch has zero output, the spurious branch must have zero parameters, i.e.,

$$
\theta_2 = 0 \quad \text{or} \quad \exists l \in \{1, ..., L-1\} \ s.t. \ \theta_2^{1(l)} = \theta_2^{2(l)} = 0
\tag{66}
$$

Thus, CIA will remove spurious features.

Now we show that when CIA objective has been reached (the spurious branch has zero outputs), the objective of $\min_\Theta \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]$ will help to learn predictive paramters of the invariant branch θ_1, $\theta_1^{1(}l)$ and $\theta_1^{2(}l)$. When Equation (66) holds,

38

$$\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]}{\partial \theta_1} = 2\mathbb{E}_e \mathbb{E}_{n_1, n_2} \left[\left(C_1^e \theta_1 - (\tilde{A}^e)^k X_1 - n_1 \right)^\top C_1^e \right]$$

$$= 2\mathbb{E}_e \left[\left(C_1^e \theta_1 - (\tilde{A}^e)^k X_1 \right)^\top C_1^e \right] \tag{67}$$

Let $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]}{\partial \theta_1} = 0$, we get the predictive parameters

$$\begin{cases} \theta_1 = 1 \\ \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, \quad l = L - 1, ..., L - k + 1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, \quad l = L - k, L - k - 1, ..., 1 \end{cases} . \tag{68}$$

Plug the final solution back in $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1^{1(l)}}$, $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1^{2(l)}}$, $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2^{1(l)}}$, $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2^{2(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]}{\partial \theta_1^{1(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]}{\partial \theta_1^{2(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]}{\partial \theta_2^{1(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e, X^e), Y^e)]}{\partial \theta_2^{2(l)}}$, we can verify that these terms are all when further letting $\exists l \in \{1, ..., L-1\}$ s.t. $\theta_2^{1(l)} = \theta_2^{2(l)} = 0$. $\qquad\square$

G.2 Proof of the Covariate Shift Case

G.2.1 Proof of the non-Graph Success Case of VREx and IRMv1 under Covariate Shift

We restate Proposition B.1 as Proposition G.5 below.

Proposition G.5. *(VREx and IRMv1 learn invariant features for non-graph tasks under covariate shift, proof is in) For the non-graph version of the SCM in Equation (19),*

$$Y^e = X_1 + n_1, \ X_2^e = n_2 + \epsilon^e, \tag{69}$$

Optimizing VREx $\min_\Theta \mathcal{L}_{VREx} = \mathbb{V}_e[R(e)]$ *and IRMv1* $\min_\Theta \mathcal{L}_{IRMv1} = \mathbb{E}_e[\|\nabla_{w|w=1.0} R(e)\|^2]$ *will learn invariant features when using a 1-layer linear network:* $f(X) = \theta_1 X_1 + \theta_2 X_2$.

Proof. **VREx.** For covariate shift, denote $X_1 \theta_1 + X_2^e \theta_2 - X_1 - n_1$ as l_e, the expected risk in environment e is $R(e) = \mathbb{E}_{n_1, n_2} \|\theta_1 X_1 + \theta_2(n_2 + \epsilon^e) - (X_1 + n_1)\|_2^2$. Take the derivative of the VREx objective $\mathbb{V}_e[R(e)]$ with respect to θ_1:

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = \mathbb{E}_e \left[2\mathbb{E}_{n_1, n_2} \left[l_e^\top l_e \right] \mathbb{E}_{n_1, n_2} \left[2l_e^\top X_1 \right] \right]$$

$$- 2\mathbb{E}_e \left[\mathbb{E}_{n_1, n_2} \left[l_e^\top l_e \right] \right] \mathbb{E}_e \left[\mathbb{E}_{n_1, n_2} \left[2l_e^\top X_1 \right] \right] \tag{70}$$

Calculate these terms:

$$\mathbb{E}_e[\mathbb{E}_{n_1, n_2}[l_e^\top l_e] \mathbb{E}_{n_1, n_2}[l_e^\top X_1]]$$
$$= (\theta_1)^3 (X_1^\top X_1)^2 + \theta_1(\theta_2)^2 (3X_1^\top X_1 \sigma^2 + X_1^\top X_1 \mathbb{E}_e[N^e]) + (\theta_2)^3 \mathbb{E}_e[(\epsilon^e)^\top \epsilon(\epsilon^e)^\top] X_1$$
$$+ (\theta_1)^2 (X_1^\top X_1 - 2(X_1^\top X_1)^2) - (\theta_2)^2 (X_1^\top X_1)^2 (3\sigma^2 + \mathbb{E}_e[N^e])$$
$$+ \theta_1 (X_1^\top X_1)^2 (3(X_1^\top X_1)^2 + \mathbb{E}_e[N^e]) - (X_1^\top X_1)^2 ((X_1^\top X_1)^2 + \mathbb{E}_e[N^e])$$
$$\mathbb{E}_e[\mathbb{E}_{n_1, n_2}[l_e^\top l_e]] \mathbb{E}_e[\mathbb{E}_{n_1, n_2}[l_e^\top X_1]]$$
$$= (\theta_1)^3 (X_1^\top X_1)^2 + \theta_1(\theta_2)^2 (X_1^\top X_1 \sigma^2 + X_1^\top X_1 \mathbb{E}_e[N^e])$$
$$+ (\theta_1)^2 (X_1^\top X_1 - 2(X_1^\top X_1)^2) - (\theta_2)^2 (X_1^\top X_1)^2 (\sigma^2 + \mathbb{E}_e[N^e])$$
$$+ \theta_1 (X_1^\top X_1)^2 (3(X_1^\top X_1)^2 + \mathbb{E}_e[N^e]) - (X_1^\top X_1)^2 ((X_1^\top X_1)^2 + \mathbb{E}_e[N^e]) \tag{71}$$

Hence,

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = \theta_1(\theta_2)^2 (2X_1^\top X_1 \sigma^2) + (\theta_2)^3 X_1 - 2(\theta_2)^2 X_1 \sigma^2 \tag{72}$$

Let $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = 0$, we have $\theta_2 = 0$.

When $\theta_2 = 0$ and when $\theta_1 = 1$, we get

$$\mathbb{E}_{n_1, n_2}[l_e^\top (n_2 + \epsilon^e)] = \theta_1 X_1^\top \epsilon - X_1^\top \epsilon^e = 0. \tag{73}$$

39

As a result,

$$
\begin{aligned}
\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2} &= \mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\mathbb{E}_{n_1,n_2}\left[2l_e^\top(n_2+\epsilon^e)\right]\right] \\
&\quad - 2\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[2l_e^\top(n_2+\epsilon^e)\right]\right] \\
&= 0 - 0 \\
&= 0
\end{aligned}
\tag{74}
$$

In conclusion, $\theta_1 = 1$ and $\theta_2 = 0$ is the solution of the VREx objective in this non-graph covariate shift task.

IRMv1. The objective of IRMv1 is $\mathbb{E}_e\|\nabla_w R(e)\|_2^2$. When IRMv1 loss is optimized to zero, we have $\nabla_w R(e) = 0$ for all environments e.

$$
\begin{aligned}
\nabla_w R(e) &= \mathbb{E}_{n_1,n_2}[2(\theta_1 X_1 + \theta_2 X_2 - (X_1 + n_1))(\theta_1 X_1 + \theta_2 X_2)] \\
&= 2((\theta_1)^2 X_1^\top X_1 + (\theta_2)^2((\epsilon^e)^\top \epsilon^e + N^e) + 2\theta_1\theta_2 X_1^\top \epsilon^e \\
&\quad - \theta_1 X_1^\top X_1 - \theta_2 X_1^\top \epsilon^e)
\end{aligned}
\tag{75}
$$

To realize $\nabla_w R(e) = 0$ for all e, we must let $\theta_2 = 0$ (and consequently $\theta_1 = 1$), otherwise the solution of θ_2 will include terms related to ϵ^e and N^e that vary with environments, and a single value θ_2 cannot fit all these values. Thus we finish the proof for IRMv1. $\qquad\square$

G.2.2 Proof of the Failure Case on Graphs of VREx under Covariate Shift

We restate Theorem B.2 as Theorem G.6 below:

Theorem G.6. *(VREx will use spurious features on graphs under covariate shift) Under the SCM of Equation (19), the objective $\min_\Theta \mathbb{V}_e[R(e)]$ has non-unique solutions for parameters of the GNN (3) when part of the model parameters $\{\theta_1^{1(l)}, \theta_1^{2(l)}, \theta_2^{1(l)}, \theta_2^{2(l)}\}$ take the values*

$$
\Theta_0 = \begin{cases}
\theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s+1 \\
\theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s, L-s-1, ..., 1 \\
\theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-1, ..., 1
\end{cases}
\tag{76}
$$

$0 < s < L$ is some positive integer, θ_1 and θ_2 have four sets of solutions of the quadratic equation, some of which are spurious solutions that $\theta_2 \neq 0$ (although $\theta_2 = 0$ is indeed one of the solutions, VREx is not guaranteed to reach this solution):

$$
\begin{cases}
c_1\sigma^2(2\theta_1\theta_2 + (\theta_2)^2 - 2c_2\sigma^2\theta_2) + c_3\theta_2 - \mathbb{E}_e[N^e]c_1\sigma^2\theta_1\theta_2 + \mathbb{E}_e[N^e]c_2\sigma^2\theta_2 = 0 \\
[c_3\theta_2 - \mathbb{E}_e[N^e]c_1\sigma^2\theta_1\theta_2 + \mathbb{E}_e[N^e]c_2\sigma^2\theta_2] c_4 - c_5(\theta_2)^2 = 0
\end{cases}.
\tag{77}
$$

where $c_1 = \mathbb{E}[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)]$, $c_2 = \mathbb{E}[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^k}X_1)]$, $c_3 = \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\sigma^2$, $c_4 = \mathbb{E}_e\left[((\tilde{A}^e)^k X_1)^\top \mathbf{1_{N^e}}\right]\sigma^2$, $c_5 = \mathbb{E}_e\left[N^e\left(tr((\tilde{A}^{e^k})^\top\tilde{A}^{e^k}) + N^e(1+\sigma^2)\right)\right]$.

Proof. We will use some symbols to simplify the expression of the toy GNN. Denote $n_2 + \epsilon^e$ as η. Use the following notations to represent the components of the L-layer GNN model:

$$
\begin{aligned}
f_\Theta(A, X) &= H_1^{(L)}\theta_1 + H_2^{(L)}\theta_2 \\
&= \underbrace{\left[\theta_1^{1(L-1)}\bar{A}\left(...\theta_1^{1(3)}\left(\theta_1^{1(2)}\bar{A}(\theta_1^{1(1)}\bar{A} + \theta_1^{2(1)}\bar{I})X_1 + \theta_1^{2(2)}(\theta_1^{1(1)}\bar{A} + \theta_1^{2(1)}\bar{I})X_1\right) + ...\right)\right]}_{C_1}\theta_1 \\
&\quad + \underbrace{\left[\theta_2^{1(L-1)}\bar{A}\left(...\theta_2^{1(3)}\left(\theta_2^{1(2)}\bar{A}(\theta_2^{1(1)}\bar{A} + \theta_2^{2(1)}\bar{I})\eta + \theta_2^{2(2)}(\theta_2^{1(1)}\bar{A} + \theta_2^{2(1)}\bar{I})\eta\right) + ...\right)\right]}_{Z}\theta_2 \\
&= C_1\theta_1 + Z\theta_2.
\end{aligned}
\tag{78}
$$

$C_1, Z \in \mathbb{R}^{N \times 1}$. We use C_1^e and Z^e to denote the variables from the corresponding environment e. We further denote $Z^e = C_2^{e\prime}\eta$.

40

Using these notations, the loss of environment e is

$$R(e) = \mathbb{E}_{n_1, n_2} \left[\|f_{\Theta}(A^e, X^e) - Y^e\|_2^2 \right]$$
$$= \mathbb{E}_{n_1, n_2} \left[\left\| C_1^e \theta_1 + Z^e \theta_2 - \tilde{A}^{e^k} X_1 - n_1 \right\|_2^2 \right]. \tag{79}$$

Denote the inner term $C_1^e \theta_1 + Z^e \theta_2 - \tilde{A}^{e^k} X_1 - n_1$ as l_e. The variance of loss across environments is:

$$\mathbb{V}_e[R(e)] = \mathbb{E}_e[R^2(e)] - \mathbb{E}_e^2[R(e)]$$
$$= \mathbb{E}_e \left[\left(\mathbb{E}_{n_1, n_2} \left\| C_1^e \theta_1 + Z^e \theta_2 - \tilde{A}^{e^k} X_1 - n_1 \right\|_2^2 \right)^2 \right]$$
$$- \mathbb{E}_e^2 \left[\mathbb{E}_{n_1, n_2} \left\| C_1^e \theta_1 + Z^e \theta_2 - \tilde{A}^{e^k} X_1 - n_1 \right\|_2^2 \right]. \tag{80}$$
$$= \mathbb{E}_e \left[\mathbb{E}_{n_1, n_2} \left[(l_e^\top l_e)^2 \right] \right] - \mathbb{E}_e^2 \left[\mathbb{E}_{n_1, n_2} \left[l_e^\top l_e \right] \right].$$

Take the derivative of $\mathbb{V}_e[R(e)]$ with respect to θ_1:

$$\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = \mathbb{E}_e \left[2\mathbb{E}_{n_1, n_2} \left[l_e^\top l_e \right] \mathbb{E}_{n_1, n_2} \left[2l_e^\top C_1^e \right] \right]$$
$$- 2\mathbb{E}_e \left[\mathbb{E}_{n_1, n_2} \left[l_e^\top l_e \right] \right] \mathbb{E}_e \left[\mathbb{E}_{n_1, n_2} \left[2l_e^\top C_1^e \right] \right] \tag{81}$$

Calculate the derivative by terms:

$$\mathbb{E}_{n_1, n_2}[l_e^\top l_e] = \mathbb{E}_{n_1, n_2}[C_1^{e\top} C_1^e (\theta_1)^2 + C_1^{e\top} Z^e \theta_1 \theta_2 - C_1^{e\top} (\tilde{A}^e)^k X_1 \theta_1 - C_1^{e\top} n_1 \theta_1$$
$$+ Z^{e\top} C_1^e \theta_1 \theta_2 + Z^{e\top} Z^e (\theta_2)^2 - Z^{e\top} (\tilde{A}^e)^k X_1 \theta_2 - Z^{e\top} n_1 \theta_2$$
$$- ((\tilde{A}^e)^k X_1)^\top C_1^e \theta_1 - ((\tilde{A}^e)^k X_1)^\top Z^e \theta_2 + ((\tilde{A}^e)^k X_1)^\top (\tilde{A}^e)^k X_1 \tag{82}$$
$$+ ((\tilde{A}^e)^k X_1)^\top n_1 - n_1^\top C_1^e \theta_1 - n_1^\top Z^e \theta_2 + n_1^\top (\tilde{A}^e)^k X_1 + n_1^\top n_1]$$

Since n_1 and n_2 are independent standard Gaussian noise, we have $\mathbb{E}_{n_1, n_2}[n_1] = \mathbb{E}_{n_1, n_2}[n_2] = \mathbf{0}$, $\mathbb{E}_{n_1, n_2}[n_1^\top n_2] = \mathbb{E}_{n_1, n_2}[n_2^\top n_1] = 0$ and $\mathbb{E}_{n_1, n_2}[n_1^\top n_1] = \mathbb{E}_{n_1, n_2}[n_2^\top n_2] = N^e$ if it is the noise from e. Also, since ϵ^e and n_1, n_2 are independent, we have $\mathbb{E}_{n_1, n_2}[n_1^\top \epsilon^e] = \mathbb{E}_{n_1, n_2}[n_2^\top \epsilon^e] = 0$.

When

$$\begin{cases} \theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s+1 \\ \theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s, L-s-1, ..., 1 \\ \theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-1, ..., 1 \end{cases} , \tag{83}$$

we have $C_2^{e\prime} = I_{N^e} \in \mathbb{R}^{N^e \times N^e}$ and $C_1^e = \tilde{A}^{e^s} X_1$. Consequently, we get $\mathbb{E}_{n_1, n_2}[Z^{e\top} n_1] = 0$, $\mathbb{E}_{n_1, n_2}[Z^{e\top} Z^e] = N^e + \epsilon^{e\top} \epsilon^e$.

Use the above conclusions and rewrite Equation (82) as (here we only plug in the value of $C_2^{e\prime}$):

$$\mathbb{E}_{n_1, n_2}[l_e^\top l_e] =$$
$$\left. \begin{array}{l} C_1^{e\top} C_1^e (\theta_1)^2 - C_1^{e\top} (\tilde{A}^e)^k X_1 \theta_1 \\ - ((\tilde{A}^e)^k X_1)^\top C_1^e \theta_1 + ((\tilde{A}^e)^k X_1)^\top (\tilde{A}^e)^k X_1 + N^e \left(1 + (\theta_2)^2 \right) \end{array} \right\} (*) \tag{84}$$
$$+ [C_1^{e\top} \epsilon^e + \epsilon^{e\top} C_1^e] \theta_1 \theta_2 + \epsilon^{e\top} \epsilon^e (\theta_2)^2 - 2((\tilde{A}^e)^k X_1)^\top \epsilon^e \theta_2 \} (**),$$

$(*)$ and $(**)$ represent terms that are independent and associated with ϵ^e, respectively.

Additionally,

$$\mathbb{E}_{n_1, n_2}[2l_e^\top C_1^e] = 2 \left[C_1^{e\top} C_1^e \theta_1 + \epsilon^{e\top} C_1^e \theta_2 - ((\tilde{A}^e)^k X_1)^\top C_1^e \right] \tag{85}$$

.

41

Multiplying Equation (84) and (85) and take the expectation on e, using the assumption that $\mathbb{E}_e[(\epsilon^e{}_i)^2] = \sigma^2$ ($\epsilon^e{}_i$ is the i-th element of ϵ^e):

$$
\mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}[l_e^\top l_e]\mathbb{E}_{n_1,n_2}\left[2l_e^\top C_1\right]\right] = 4\mathbb{E}_e\left[(*)\left(C_1^{e\top}C_1^e\theta_1 - ((\tilde{A}^e)^k X_1)^\top C_1^e\right)\right]
$$
$$
+ 4\mathbb{E}_e\left[(\tilde{A}^{e^s}X_1)^\top \tilde{A}^{e^s}X_1(2\theta_1\theta_2 + (\theta_2)^2) - 2(\tilde{A}^{e^s}X_1)^\top((\tilde{A}^e)^k X_1)\theta_2\right]\theta_2\sigma^2
$$
$$
+ 4\mathbb{E}_e[N^e\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\theta_2.
$$
(86)

Next target is to compute $2\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[l_e^\top l_e]]$ and $\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[2l_e^\top C_1]]$ Since ϵ^e has zero mean, we have:

$$
2\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[l_e^\top l_e]] = \mathbb{E}[(*)] + \mathbb{E}[2N^e]\sigma^2(\theta_2)^2
$$
(87)

and

$$
\mathbb{E}_e[\mathbb{E}_{n_1,n_2}[2l_e^\top C_1^e]] = 2\mathbb{E}_e\left[C_1^{e\top}C_1^e\theta_1 - ((\tilde{A}^e)^k X_1)^\top C_1^e\right].
$$
(88)

Use Equation (86) (87) and (88) and let $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = 0$, we have:

$$
\mathbb{E}_e\left[2(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)(\theta_1\theta_2 + \frac{1}{2}(\theta_2)^2) - 2(\tilde{A}^{e^s}X_1)^\top((\tilde{A}^e)^k X_1)\theta_2\right]\sigma^2 + \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\theta_2
$$
$$
- \mathbb{E}_e[N^e]\mathbb{E}_e\left[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)\theta_1 - ((\tilde{A}^e)^k X_1)^\top C_1^e\right]\theta_2\sigma^2 = 0.
$$
(89)

Now we start calculating the expression of $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2}$:

$$
\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2} = \mathbb{E}_e\left[2\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\mathbb{E}_{n_1,n_2}\left[2l_e^\top Z^e\right]\right]
$$
$$
- 2\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[l_e^\top l_e\right]\right]\mathbb{E}_e\left[\mathbb{E}_{n_1,n_2}\left[2l_e^\top Z^e\right]\right].
$$
(90)

Let $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2} = 0$:

$$
\mathbb{E}_e\left[(C_1^{e\top}C_2^{e\prime} + C_2^{e\prime\top}C_1^{e\top})\theta_1\theta_2 + (C_2^{e\prime})^\top C_2^{e\prime}(\theta_2)^2 - 2((\tilde{A}^e)^k X_1)^\top C_2^{e\prime}\theta_2\right]\mathbb{E}_e\left[(-((\tilde{A}^e)^k X_1)^\top C_2^{e\prime})\right]\sigma^2
$$
$$
- \mathbb{E}_e\left[N^e\sigma^2\left(\text{tr}((\tilde{A}^{e^k})^\top \tilde{A}^{e^k}) + N^e + C_2^{e\prime\top}C_2^{e\prime}\sigma^2\right)(\theta_2)^2\right]
$$
$$
= 0.
$$
(91)

Plug Equation (89) in (91), we reach:

$$
\left[\mathbb{E}_e\left[N^e(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)\theta_1 - N^e((\tilde{A}^e)^k X_1)^\top C_1^e\right]\theta_2\sigma^2 - \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\theta_2\right]\mathbb{E}_e\left(-(\tilde{A}^{e^k}X_1)^\top \mathbf{1}_{N^e}\right)
$$
$$
- \mathbb{E}_e\left[N^e\left(\text{tr}((\tilde{A}^{e^k})^\top \tilde{A}^{e^k}) + N^e(1+\sigma^2)\right)\right](\theta_2)^2
$$
$$
= 0.
$$
(92)

Let $c_1 = \mathbb{E}[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^s}X_1)]$, $c_2 = \mathbb{E}[(\tilde{A}^{e^s}X_1)^\top(\tilde{A}^{e^k}X_1)]$, $c_3 = \mathbb{E}_e[\epsilon^{e\top}\epsilon^e\epsilon^{e\top}(\tilde{A}^{e^s}X_1)]\sigma^2$, $c_4 = \mathbb{E}_e\left[((\tilde{A}^e)^k X_1)^\top \mathbf{1}_{N^e}\right]\sigma^2$, $c_5 = \mathbb{E}_e\left[N^e\left(\text{tr}((\tilde{A}^e)^\top \tilde{A}^{e^k}) + N^e(1+\sigma^2)\right)\right]$,

we conclude that

$$
\begin{cases}
c_1\sigma^2(2\theta_1\theta_2 + (\theta_2)^2 - 2c_2\sigma^2\theta_2) + c_3 - \mathbb{E}_e[N^e]c_1\sigma^2\theta_1\theta_2 + \mathbb{E}_e[N^e]c_2\sigma^2\theta_2 = 0 \\
\left[c_3\theta_2 - \mathbb{E}_e[N^e]c_1\sigma^2\theta_1\theta_2 + \mathbb{E}_e[N^e]c_2\sigma^2\theta_2\right]c_4 - c_5(\theta_2)^2 = 0
\end{cases}.
$$
(93)

As for the derivative respect to $\theta_1^{1(l)}, \theta_1^{2(l)}, \theta_2^{1(l)}, \theta_2^{2(l)}$, when they take the special value in Equation (83), we have $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1} = 0 \Rightarrow \frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1^{1(l)}} = \frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_1^{2(l)}} = 0$ and $\frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2} = 0 \Rightarrow \frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2^{1(l)}} = \frac{\partial \mathbb{V}_e[R(e)]}{\partial \theta_2^{2(l)}} = 0, l = 1, ..., L$. So we conclude the solution induced by Equation (93) is the solution of the objective, and $\theta_2 = 0$ is not a valid solution.

$\square$

42

G.2.3 Proof of the Failure Case on Graphs of IRMv1 under Covariate Shift

We restate Theorem B.3 as Theorem G.7 below:

Theorem G.7. *(**IRMv1 will use spurious features on graphs under covariate shift**) Under the SCM of Equation (19), there exists $s \in \mathbb{N}^+$ that satisfies $0 < s < L$ and $s \neq k$ such that optimizing the IRMv1 objective $\min_\Theta \mathbb{E}_e[\|\nabla_{w|w=1.0} R(e)\|^2]$ will not lead to the invariant solution $\theta_2 = 0$ for parameters of the GNN (3) when $\{\theta_1^{1(l)}, \theta_1^{2(l)}, \theta_2^{1(l)}, \theta_2^{2(l)}\}$ take the special solution:*

$$
\Theta_0 = \begin{cases}
\theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-s+1 \\
\theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-s, L-s-1, ..., 1 \\
\theta_2^{1(l)} = 0, \theta_2^{2(l)} = 1, & l = L-1, ..., 1
\end{cases} . \tag{94}
$$

Proof. From the proof of non-graph IRMv1 case Appendix G.1.1 we know that when the IRMv1 objective is optimized, we have $\nabla_w R(e) = 0$ for all e. For the graph case, the expected risk of environment e is

$$
R(e) = \mathbb{E}_{n_1, n_2}[\|\theta_1 C_1^e + \theta_2 Z^e - \tilde{A}_e^k X_1 - n_1\|_2^2], \tag{95}
$$

where the definition of C_1^e and Z^e follows Equation (78). Now let's check if the invariant solution $\theta_2 = 0$ is a valid solution. Let $\theta_2 = 0$,

$$
\begin{aligned}
\nabla_w R(e) &= \mathbb{E}_{n_1, n_2}[(\theta_1 C_1^e + \theta_2 Z^e - (\tilde{A}^e)^k X_1 - n_1)^\top (\theta_1 C_1^e + \theta_2 Z^e)] \\
&= (\theta_1)^2 ((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^s X_1) + (\theta_2)^2 [N^e + (\epsilon^e)^\top \epsilon^e] + 2\theta_1 \theta_2 ((\tilde{A}^e)^s X_1)^\top \epsilon \\
&\quad - \theta_1 ((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^k X_1) - \theta_2 (N^e + (\epsilon^e)^\top)((\tilde{A}^e)^k X_1)
\end{aligned} \tag{96}
$$

If $\theta_2 = 0$ is a feasible solution, then the following equation must hold for every environment e:

$$
(\theta_1)^2 ((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^s X_1) - \theta_1 ((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^k X_1) = 0. \tag{97}
$$

When $s \neq k$, we get $\theta_1 = \frac{((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^s X_1)}{((\tilde{A}^e)^s X_1)^\top ((\tilde{A}^e)^k X_1)}$ (we discard the trivial solution of $\theta_1 = 0$). The value of this solution of θ_1 varies with environment e, and thus is not a valid solution. However, now we will show that optimizing IRMv1 does not necessarily lead to lower-layer parameters such that $s = k$. To reveal this, by taking the derivative of $\mathcal{L}_{\text{IRMv1}}$ w.r.t. θ_1 and θ_2 and let them $= 0$, we can get two cubic equations:

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1} =& \mathbb{E}_e[(C_1^{e\top} C_1^e (\theta_1)^2 + (N^e + \epsilon^\top \epsilon)(\theta_2)^2 + C_1^{e\top} \epsilon^e \theta_1 \theta_2 - (\tilde{A}^{e^k} X_1)^\top C_1^e \theta_1) \\
& (2 C_1^{e\top} C_1^{e\top} \theta_1 + C_1^{e\top} \epsilon \theta_2 - (\tilde{A}^{e^k} X_1)^\top C_1^e)] = 0
\end{aligned} \tag{98}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2} =& \mathbb{E}_e[(C_1^{e\top} C_1^e (\theta_1)^2 + (N^e + \epsilon^\top \epsilon)(\theta_2)^2 + C_1^{e\top} \epsilon^e \theta_1 \theta_2 - (\tilde{A}^{e^k} X_1)^\top C_1^e \theta_1) \\
& (2 (N^e + \epsilon^{e\top} \epsilon^e) \theta_2 + (C_1^{e\top} C_2^e + C_1^{e\top} \epsilon) \theta_1)] = 0
\end{aligned} \tag{99}
$$

From the analysis in Appendix F.3, we know that as long as the lower-layer parameters take any value that satisfies the form in Equation (27), even if $s \neq k$, we can get $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^2}$ and $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^2}$. At this time (when $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_1^2} = 0$ and $\frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^1} = \frac{\partial \mathcal{L}_{\text{IRMv1}}}{\partial \theta_2^2} = 0$ but $s \neq k$), from the form of Equations (98) and (99) we know that there exist solutions that $\theta_2 \neq 0$, and the solution of θ_1 and θ_2 both depend on $\mathbb{E}_e(F(e))$, where $F(e)$ is some random variable associated with e.

□

G.2.4 Proof of the Successful Case on Graphs of CIA under Covariate Shift

Theorem G.8. *Optimizing the CIA objective will lead to the optimal solution Θ^*:*

$$
\begin{cases}
\theta_1 = 1 \\
\theta_2 = 0 \\
\theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-k+1 \\
\theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-k, L-k-1, ..., 1
\end{cases} . \tag{100}
$$

43

Proof. For brevity, denote a node representation of $C_{1_c}^e$ as C_1^i and the one of $C_{1_c}^{e'}$ as C_1^j. The same is true for C_2^i and C_2^j. In this toy model, we need to consider the expectation of the noise, while in real cases such noise is included in the node features so taking expectation on e will handle this. Therefore, we add $\mathbb{E}_{n_1,n_2}$ in this proof, and this expectation is excluded in the formal description of the objective in the main paper.

$$
\begin{aligned}
\mathcal{L}_{\text{CIA}}c &= \mathbb{E}_{\substack{e,e' \\ e\neq e'}} \mathbb{E}_{n_1,n_2} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j)\in\Omega^{e,e'}}} \left[\mathcal{D}(\phi_\Theta(A^e,X^e)_{[c][v_i]}, \phi_\Theta(A^e,X^{e'})_{[c][v_j]}) \right] \\
&= \mathbb{E}_{\substack{e,e' \\ e\neq e'}} \mathbb{E}_{n_1,n_2} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j)\in\Omega^{e,e'}}} \| C_1^i + Z^e - C_1^j - Z^{e'} \|_2^2
\end{aligned}
\tag{101}
$$

$$
\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1} = \mathbb{E}_{\substack{e,e' \\ e\neq e'}} \mathbb{E}_{n_1,n_2} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j)\in\Omega^{e,e'}}} \left[C_1^i\theta_1 + Z^e\theta_2 - C_1^j\theta_1 - Z^{e'}\theta_2 \right]^\top (C_1^i - C_1^j)
\tag{102}
$$

Let $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1} = 0$, we have:

$$
\mathbb{E}_{\substack{e,e' \\ e\neq e'}} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j)\in\Omega^{e,e'}}} \left[(C_1^i - C_1^j)^\top (C_1^i - C_1^j)\theta_1 \right] = 0
\tag{103}
$$

Thus, we get two possible solutions of the invariant branch. The first valid solution is the optimal one:

$$
\begin{cases}
\theta_1 = 1 \\
\theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-k+1 \\
\theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-k, L-k-1, ..., 1
\end{cases}
\tag{104}
$$

The second valid solution is a trivial one:

$$
\theta_1 = 0 \quad \text{or} \quad \exists l \in \{1,...,L-1\} \text{ s.t. } \theta_1^{1(l)} = \theta_1^{2(l)} = 0
\tag{105}
$$

Take the derivative of the objective w.r.t. θ_2:

$$
\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2} = \mathbb{E}_{\substack{e,e' \\ e\neq e'}} \mathbb{E}_c \mathbb{E}_{\substack{i,j \\ (i,j)\in\Omega^{e,e'}}} \left[\left[(Z^e - Z^{e'})^\top (Z^e - Z^{e'}) \right] \theta_2 \right] = 2\sigma^2\theta_2
\tag{106}
$$

Let $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2} = 0$, we get $\theta_2 = 0$. Thus, CIA will remove spurious features.

Now we show that when CIA objective has been reached (the spurious branch has zero outputs), the objective of $\min_\Theta \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]$ will help to learn predictive parameters of the invariant branch θ_1, $\theta_1^{1}(l)$ and $\theta_1^{2}(l)$). When $\theta_2 = 0$:

$$
\begin{aligned}
\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]}{\partial \theta_1} &= 2\mathbb{E}_e\mathbb{E}_{n_1,n_2} \left[\left(C_1^e\theta_1 - (\tilde{A}^e)^k X_1 - n_1 \right)^\top C_1^e \right] \\
&= 2\mathbb{E}_e \left[\left(C_1^e\theta_1 - (\tilde{A}^e)^k X_1 \right)^\top C_1^e \right]
\end{aligned}
\tag{107}
$$

Let $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]}{\partial \theta_1} = 0$, we get the predictive parameters

$$
\begin{cases}
\theta_1 = 1 \\
\theta_1^{1(l)} = 1, \theta_1^{2(l)} = 1, & l = L-1, ..., L-k+1 \\
\theta_1^{1(l)} = 0, \theta_1^{2(l)} = 1, & l = L-k, L-k-1, ..., 1
\end{cases}
\tag{108}
$$

Plug the final solution back in $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1^{1(l)}}$, $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_1^{2(l)}}$, $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2^{1(l)}}$, $\frac{\partial \mathcal{L}_{\text{CIA}}}{\partial \theta_2^{2(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]}{\partial \theta_1^{1(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]}{\partial \theta_1^{2(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]}{\partial \theta_2^{1(l)}}$, $\frac{\partial \mathbb{E}_e[\mathcal{L}(f_\Theta(A^e,X^e),Y^e)]}{\partial \theta_2^{2(l)}}$, we can verify that these terms are all 0.

$\square$

44

G.3 Proof of Theorem 4.4

The following proof of Theorem 4.4 is adapted from Ma et al. [2021] and Mao et al. [2023]. We restate Theorem 4.4 as Theorem G.9 below.

Theorem G.9. *(Subgroup Generalization Bound for GNNs). Let $\tilde{h}$ be any classifier in a function family $\mathcal{H}$ with parameters $\left\{\widetilde{W}_l\right\}_{l=1}^{L}$. Under Assumption 4.2 and 4.3, for any $e^{te} \in \mathcal{E}_{te}, \gamma \geq 0$, and large enough $N_{e^{tr}}$, with probability at least $1 - \delta$, we have*

$$
\begin{aligned}
\mathcal{L}_{e^{te}}^{0}(\tilde{h}) \leq & \widehat{\mathcal{L}}_{e^{tr}}^{\gamma}(\tilde{h}) \\
& + O(\frac{1}{\sigma^2}(\underbrace{\sum_{c=1}^{C}\sum_{c'\neq c}(\sqrt{|[(\mu_c - \mu_{c'})^\top ; (\mu_c^{e^{te}} - \mu_{c'}^{e^{te}})^\top]|} + 2\sqrt{2})\epsilon_{e^{te},e^{tr}}}_{(a)} \\
& + 2\underbrace{\sum_{c=1}^{C}(C-1)B_{e^{te}}|\mu_c^{e^{te}} - \mu_c^{e^{tr}}|)}_{(b)} \\
& + \underbrace{\frac{1}{2\sigma^2}\frac{1}{N_{e^{tr}}}\sum_{i\in V_{e^{tr}}}\frac{1}{N_{e^{te}}}\sum_{j\in V_{e^{te}}^{(i)}}\sum_{c=1}^{C}\sum_{c'\neq c}|p_j^{ht}(c'|c) - p_i^{ht}(c'|c)|}_{(c)} \\
& + \underbrace{\frac{b\sum_{l=1}^{L}\left\|\widetilde{W}_l\right\|_F^2}{(\gamma/8)^{2/L}N_{e^{tr}}^{\alpha}}\left(\epsilon_{e^{te},e^{tr}}\right)^{2/L}}_{(d)} + const)
\end{aligned}
\tag{109}
$$

where $p_i^{ht}(c'|c)$ is the ratio of heterophilic neighbors of class c' when $y_i = c$, $const = \frac{1}{N^{1-2\alpha}} + \frac{1}{N_{e^{tr}}^{2\alpha}}\ln\frac{LC(2B_{e^{te}})^{1/L}}{\gamma^{1/L}\delta}$ is a term independent of aggregated feature distance and the difference in neighboring heterophilic label distribution, where $B_{e^{te}} = \max_{i\in V_{e^{tr}}\cup V_{e^{te}}}\|g_i\|_2$ is the maximum feature norm.

To prove Theorem G.9, we need the following lemma that bounds the expected loss discrepancy between the train and the test node subgroups.

Lemma G.10. *Given a distribution P over $\mathcal{H}$, for $\lambda > 0$, define the expected loss discrepancy between V_e and $V_{e'}$ with respect to (P, γ, λ) as $D_{e,e'}^{\gamma}(P;\lambda) := \ln\mathbb{E}_{h\sim P}e^{\lambda(\mathcal{L}_e^{\gamma/2}(h) - \mathcal{L}_{e'}^{\gamma}(h))}$. Under Assumption 4.2, we have*

$$
\begin{aligned}
D_{e^{te},e^{tr}}^{\gamma/2} \leq & \frac{1}{\sigma^2}(\sum_{c=1}^{C}\sum_{c'\neq c}(\sqrt{|[(\mu_c - \mu_{c'})^\top ; (\mu_c^{e^{te}} - \mu_{c'}^{e^{te}})^\top]|} + 2\sqrt{2})\epsilon_{e^{te},e^{tr}} + 2\sum_{c=1}^{C}(C-1)B_{e^{te}}|\mu_c^{e^{te}} - \mu_c^{e^{tr}}|) \\
& + \frac{1}{2\sigma^2}\frac{1}{N_{e^{tr}}}\sum_{i\in V_{e^{tr}}}\frac{1}{N_{e^{te}}}\sum_{j\in V_{e^{te}}^{(i)}}\sum_{c=1}^{C}\sum_{c'\neq c}|p_j^{ht}(c'|c) - p_i^{ht}(c'|c)|
\end{aligned}
\tag{110}
$$

45

Proof. According to Eq. (14) of Lemma 5 in Ma et al. [2021], under 4.2, we already have

$$\mathcal{L}_{e^{\text{te}}}^0(\tilde{h}) - \widehat{\mathcal{L}}_{e^{\text{tr}}}^\gamma(\tilde{h})$$

$$\leq \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^{C} \left(1 \cdot \left(\mathcal{L}^{\frac{\gamma}{2}}(h_j, c) - \mathcal{L}^\gamma(h_i, k) \right) + (\Pr(y_j = c|g_j) - \Pr(y_i = c|g_i)) \cdot 1 \right)$$

$$\leq \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^{C} (\Pr(y_j = c|g_j) - \Pr(y_i = c|g_i))$$

$$\tag{111}$$

In the following proof, the main goal is to bound the term $(\Pr(y_j = c|g_j) - \Pr(y_i = c|g_i))$ under the multi-classification OOD generalization setting. Using the Bayes theorem, we have

$$\frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^{C} (\Pr(y_j = c|g_j) - \Pr(y_i = c|g_i))$$

$$= \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^{C} \frac{\Pr(g_j|y_j = c) \sum_{c' \neq c} \Pr(g_i|y_j = c') - \Pr(g_i|y_i = c) \sum_{c' \neq c} \Pr(g_j|y_j = c')}{(\sum_{c'=1}^{C} \Pr(g_j|y_j = c'))(\sum_{c'=1}^{C} \Pr(g_i|y_i = c'))}$$

$$\tag{112}$$

Let consider the term $\frac{\Pr(g_j|y_j=c)\Pr(g_i|y_j=c') - \Pr(g_i|y_i=c)\Pr(g_j|y_j=c')}{(\sum_{c'=1}^{C} \Pr(g_j|y_j=c'))(\sum_{c'=1}^{C} \Pr(g_i|y_i=c'))}$ for each $c' \neq c$, and bound all these terms respectively. When $y_i = c$, denote $\mu_i(c) \in \mathcal{R}^{D \times (C-1)}$ as the matrix composed of all the class means of node v_i's heterophilic neighbors, i.e. the columns of $\mu_i(c)$ are μ_c', $c' \in \{1, 2, ..., C\}/\{c\}$. The elements of $p_i^{\text{ht}} \in \mathcal{R}^{C-1}$ are corresponding $p_i^{\text{ht}}(c')$, $c' \in \{1, 2, ..., C\}/\{c\}$. According to Definition 4.1, we have

$$\Pr(g_i|y_i = c) = \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp(-\frac{1}{2\sigma^2} \|g_i - [p_i^{\text{hm}}\mu_c^\top + (\mu_i(c)p_i^{\text{ht}})^\top ; p_i^{\text{hm}}\mu_c^{e^{\text{te}}\top} + (\mu_i(c)^{e^{\text{te}}}p_i^{\text{ht}})^\top]^\top \|_2^2)$$

$$\tag{113}$$

Denote $(\sum_{c'=1}^{C} \Pr(g_j|y_j = c'))(\sum_{c'=1}^{C} \Pr(g_i|y_i = c'))$ as $\exp(M) \in [0, C]$, where $M > 0$ is a constant, we have

$$|\frac{\Pr(g_j|y_j = c)\Pr(g_i|y_j = c') - \Pr(g_i|y_i = c)\Pr(g_j|y_j = c')}{(\sum_{c'=1}^{C} \Pr(g_j|y_j = c'))(\sum_{c'=1}^{C} \Pr(g_i|y_i = c'))}|$$

$$= |\exp(-\frac{1}{2\sigma^2} \|g_j - [p_j^{\text{hm}}\mu_c^\top + (\mu_i(c)p_j^{\text{ht}})^\top ; p_j^{\text{hm}}\mu_c^{e^{\text{te}}\top} + (\mu_j(c)^{e^{\text{te}}}p_j^{\text{ht}})^\top]^\top \|_2^2)$$

$$\exp(-\frac{1}{2\sigma^2} \|g_i - [p_i^{\text{hm}}\mu_{c'}^\top + (\mu_i(c')p_i^{\text{ht}})^\top ; p_i^{\text{hm}}\mu_{c'}^{e^{\text{tr}}\top} + (\mu_i(c')^{e^{\text{tr}}}p_i^{\text{ht}})^\top]^\top \|_2^2)$$

$$\tag{114}$$

$$- \exp(-\frac{1}{2\sigma^2} \|g_j - [p_j^{\text{hm}}\mu_{c'}^\top + (\mu_i(c')p_j^{\text{ht}})^\top ; p_j^{\text{hm}}\mu_{c'}^{e^{\text{te}}\top} + (\mu_j(c')^{e^{\text{te}}}p_j^{\text{ht}})^\top]^\top \|_2^2)$$

$$\exp(-\frac{1}{2\sigma^2} \|g_i - [p_i^{\text{hm}}\mu_c^\top + (\mu_i(c)p_i^{\text{ht}})^\top ; p_i^{\text{hm}}\mu_c^{e^{\text{tr}}\top} + (\mu_i(c)^{e^{\text{tr}}}p_i^{\text{ht}})^\top]^\top \|_2^2)| / |\exp(-M)|$$

Further arranging the formula, we obtain:

$$|\frac{\Pr(g_j|y_j = c)\Pr(g_i|y_j = c') - \Pr(g_i|y_i = c)\Pr(g_j|y_j = c')}{(\sum_{c'=1}^{C} \Pr(g_j|y_j = c'))(\sum_{c'=1}^{C} \Pr(g_i|y_i = c'))}|$$

$$= |\exp(-\frac{\|g_j - [p_j^{\text{hm}}\mu_c^\top + (\mu_i(c)p_j^{\text{ht}})^\top ; p_j^{\text{hm}}\mu_c^{e^{\text{te}}\top} + (\mu_j(c)^{e^{\text{te}}}p_j^{\text{ht}})^\top]^\top \|_2^2}{2\sigma^2}$$

$$- \frac{\|g_i - [p_i^{\text{hm}}\mu_{c'}^\top + (\mu_i(c')p_i^{\text{ht}})^\top ; p_i^{\text{hm}}\mu_{c'}^{e^{\text{tr}}\top} + (\mu_i(c')^{e^{\text{tr}}}p_i^{\text{ht}})^\top]^\top \|_2^2}{2\sigma^2} - M) \tag{115}$$

$$- \exp(-\frac{\|g_j - [p_j^{\text{hm}}\mu_{c'}^\top + (\mu_i(c')p_j^{\text{ht}})^\top ; p_j^{\text{hm}}\mu_{c'}^{e^{\text{te}}\top} + (\mu_j(c')^{e^{\text{te}}}p_j^{\text{ht}})^\top]^\top \|_2^2}{2\sigma^2}$$

$$- \frac{\|g_i - [p_i^{\text{hm}}\mu_c^\top + (\mu_i(c)p_i^{\text{ht}})^\top ; p_i^{\text{hm}}\mu_c^{e^{\text{tr}}\top} + (\mu_i(c)^{e^{\text{tr}}}p_i^{\text{ht}})^\top]^\top \|_2^2}{2\sigma^2} - M)|$$

46

Using Lagrange mean value theorem, we have

$$|\exp(-x) - \exp(-y)| = \exp(-\xi)|y - x| \le |x - y| \tag{116}$$

for x, $y > 0$ and $\xi \in [x, y]$ (or $\xi \in [y, x]$). Using Equation (116), we get

$$
\begin{aligned}
&\left| \frac{\Pr(g_j|y_j = c)\Pr(g_i|y_i = c') - \Pr(g_i|y_i = c)\Pr(g_j|y_j = c')}{(\sum_{c'=1}^{C} \Pr(g_j|y_j = c'))(\sum_{c'=1}^{C} \Pr(g_i|y_i = c'))} \right| \\
&\le \frac{1}{2\sigma^2}(\|g_j - [p_j^{\mathrm{hm}}\mu_c^\top + (\mu_i(c)p_j^{\mathrm{ht}})^\top; p_j^{\mathrm{hm}}\mu_c^{e^{\mathrm{te}}\top} + (\mu_j(c)^{e^{\mathrm{te}}}p_j^{\mathrm{ht}})^\top]^\top\|_2^2 \\
&+ \|g_i - [p_i^{\mathrm{hm}}\mu_{c'}^\top + (\mu_i(c')p_i^{\mathrm{ht}})^\top; p_i^{\mathrm{hm}}\mu_{c'}^{e^{\mathrm{tr}}\top} + (\mu_i(c')^{e^{\mathrm{tr}}}p_i^{\mathrm{ht}})^\top]^\top\|_2^2 \\
&- \|g_j - [p_j^{\mathrm{hm}}\mu_{c'}^\top + (\mu_i(c')p_j^{\mathrm{ht}})^\top; p_j^{\mathrm{hm}}\mu_{c'}^{e^{\mathrm{te}}\top} + (\mu_j(c')^{e^{\mathrm{te}}}p_j^{\mathrm{ht}})^\top]^\top\|_2^2 \\
&- \|g_i - [p_i^{\mathrm{hm}}\mu_c^\top + (\mu_i(c)p_i^{\mathrm{ht}})^\top; p_i^{\mathrm{hm}}\mu_c^{e^{\mathrm{tr}}\top} + (\mu_i(c)^{e^{\mathrm{tr}}}p_i^{\mathrm{ht}})^\top]^\top\|_2^2) \\
&= \left.\begin{aligned}
&-\tfrac{1}{\sigma^2}([(\mu_c - \mu_{c'})^\top; (\mu_c^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{te}}})^\top](g_j p_j^{\mathrm{hm}} - g_i p_i^{\mathrm{hm}}) + [0; (\mu_c^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{te}}})^\top - (\mu_c^{e^{\mathrm{tr}}} - \mu_{c'}^{e^{\mathrm{tr}}})^\top]g_i p_i^{\mathrm{hm}} \\
&+ [\mu_c^\top; \mu_c^{e^{\mathrm{tr}}\top}](p_i^{\mathrm{ht}}(c)g_i - p_j^{\mathrm{ht}}(c)g_j) + [0; \mu_c^{e^{\mathrm{tr}}\top}]p_j^{\mathrm{ht}}(c)g_j \\
&- [\mu_{c'}^\top; \mu_{c'}^{e^{\mathrm{tr}}\top}](p_i^{\mathrm{ht}}(c')g_i - p_j^{\mathrm{ht}}(c')g_j) - [0; \mu_{c'}^{e^{\mathrm{tr}}\top}]p_j^{\mathrm{ht}}(c')g_j \\
&- [0; \mu_c^{e^{\mathrm{te}}\top}]p_j^{\mathrm{ht}}(c)g_j + [0; \mu_{c'}^{e^{\mathrm{te}}\top}]p_j^{\mathrm{hm}}(c')g_j)
\end{aligned}\right\} \text{(A)} \\
&+ \tfrac{1}{2\sigma^2}((p_j^{\mathrm{hm}} - p_i^{\mathrm{hm}})(\|\mu_c\|_2^2 - \|\mu_{c'}\|_2^2) + p_j^{\mathrm{hm}}\|\mu_c^{e^{\mathrm{te}}}\|_2^2 - p_i^{\mathrm{hm}}\|\mu_c^{e^{\mathrm{tr}}}\|_2^2 + p_i^{\mathrm{hm}}\|\mu_{c'}^{e^{\mathrm{tr}}}\|_2^2 - p_j^{\mathrm{hm}}\|\mu_{c'}^{e^{\mathrm{te}}}\|_2^2) \Big\} \text{(B)} \\
&+ \left.\begin{aligned}
&\tfrac{1}{2\sigma^2}((p_j^{\mathrm{ht}}(c)^2 - p_i^{\mathrm{ht}}(c)^2)\|\mu_c\|_2^2 + p_j^{\mathrm{ht}}(c)^2\|\mu_c^{e^{\mathrm{te}}}\|_2^2 - p_i^{\mathrm{ht}}(c)^2\|\mu_c^{e^{\mathrm{tr}}}\|_2^2 \\
&- (p_j^{\mathrm{ht}}(c')^2 - p_i^{\mathrm{ht}}(c')^2)\|\mu_{c'}\|_2^2 - p_j^{\mathrm{ht}}(c')^2\|\mu_{c'}^{e^{\mathrm{te}}}\|_2^2 + p_i^{\mathrm{ht}}(c')^2\|\mu_{c'}^{e^{\mathrm{te}}}\|_2^2)
\end{aligned}\right\} \text{(C)}
\end{aligned}
\tag{117}
$$

Let's bound (A), (B) and (C) respectively.

$$
\begin{aligned}
\text{(A)} &\overset{(a)}{\le} \frac{1}{\sigma^2}(|[(\mu_c - \mu_{c'})^\top; (\mu_c^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{te}}})^\top]|\|g_i - g_j\|_2 + (|\mu_c^{e^{\mathrm{te}}} - \mu_c^{e^{\mathrm{tr}}}| + |\mu_{c'}^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{tr}}}|)|g_i| \\
&\quad + |[\mu_c^\top; \mu_c^{e^{\mathrm{tr}}\top}]|\|g_i - g_j\|_2 + |[\mu_{c'}^\top; \mu_{c'}^{e^{\mathrm{tr}}\top}]|\|g_i - g_j\|_2 \\
&\quad + (|\mu_c^{e^{\mathrm{te}}} - \mu_c^{e^{\mathrm{tr}}}| + |\mu_{c'}^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{tr}}}|)|g_j|) \\
&\overset{(b)}{=} \frac{1}{\sigma^2}\left((\sqrt{|[(\mu_c - \mu_{c'})^\top; (\mu_c^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{te}}})^\top]|} + 2\sqrt{2})\|g_i - g_j\|_2 + (|\mu_c^{e^{\mathrm{te}}} - \mu_c^{e^{\mathrm{tr}}}| + |\mu_{c'}^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{tr}}}|)(|g_i| + |g_j|)\right) \\
&\le \frac{1}{\sigma^2}\left((\sqrt{|[(\mu_c - \mu_{c'})^\top; (\mu_c^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{te}}})^\top]|} + 2\sqrt{2})\|g_i - g_j\|_2 + 2B_{e^{\mathrm{te}}}(|\mu_c^{e^{\mathrm{te}}} - \mu_c^{e^{\mathrm{tr}}}| + |\mu_{c'}^{e^{\mathrm{te}}} - \mu_{c'}^{e^{\mathrm{tr}}}|)\right)
\end{aligned}
\tag{118}
$$

(a) uses the fact that $|g_j p_j^{\mathrm{hm}} - g_i p_i^{\mathrm{hm}}| \le \|g_i - g_j\|_2$ and p_i^{hm}, $p_i^{\mathrm{ht}}(c)$, $p_i^{\mathrm{ht}}(c')$, p_j^{hm}, $p_j^{\mathrm{ht}}(c)$, $p_j^{\mathrm{ht}}(c')$ are all < 1, and (b) is because of the assumption that $|[\mu_c^\top; \mu_c^{e^{\mathrm{tr}}\top}]| = |[\mu_{c'}^\top; \mu_{c'}^{e^{\mathrm{tr}}\top}]| = \sqrt{2}$.
Also, since the class means are normalized, we have (B) = 0.
Similarly, we have

$$
\begin{aligned}
\text{(C)} &= \frac{1}{2\sigma^2}((p_j^{\mathrm{ht}}(c)^2 - p_i^{\mathrm{ht}}(c)^2) - (p_j^{\mathrm{ht}}(c')^2 - p_i^{\mathrm{ht}}(c')^2)) \\
&\le \frac{1}{2\sigma^2}\left(|p_j^{\mathrm{ht}}(c) - p_i^{\mathrm{ht}}(c)| + |p_j^{\mathrm{ht}}(c') - p_i^{\mathrm{ht}}(c')|\right)
\end{aligned}
\tag{119}
$$

47

Using these results, we have

$$\mathcal{L}_{e^{\text{te}}}^0(\tilde{h}) - \widehat{\mathcal{L}}_{e^{\text{tr}}}^\gamma(\tilde{h})$$

$$\leq \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^C \frac{\Pr(g_j|y_j = c) \sum_{c' \neq c} \Pr(g_i|y_j = c') - \Pr(g_i|y_i = c) \sum_{c' \neq c} \Pr(g_j|y_j = c')}{(\sum_{c'=1}^C \Pr(g_j|y_j = c'))(\sum_{c'=1}^C \Pr(g_i|y_i = c'))}$$

$$= \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^C \sum_{c' \neq c} \frac{1}{\sigma^2}((\sqrt{|[(\mu_c - \mu_{c'})^\top; (\mu_c^{e^{\text{te}}} - \mu_{c'}^{e^{\text{te}}})^\top]|} + 2\sqrt{2}) \|g_i - g_j\|_2)$$

$$+ \frac{2}{\sigma^2} \sum_{c=1}^C (C-1) B_{e^{\text{te}}} |\mu_c^{e^{\text{te}}} - \mu_c^{e^{\text{tr}}}|$$

$$+ \frac{1}{2\sigma^2} \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^C \sum_{c' \neq c} |p_j^{\text{ht}}(c'|c) - p_i^{\text{ht}}(c'|c)|$$

$$\leq \frac{1}{\sigma^2} (\sum_{c=1}^C \sum_{c' \neq c} (\sqrt{|[(\mu_c - \mu_{c'})^\top; (\mu_c^{e^{\text{te}}} - \mu_{c'}^{e^{\text{te}}})^\top]|} + 2\sqrt{2}) \epsilon_{e^{\text{te}}, e^{\text{tr}}} + 2 \sum_{c=1}^C (C-1) B_{e^{\text{te}}} |\mu_c^{e^{\text{te}}} - \mu_c^{e^{\text{tr}}}|)$$

$$+ \frac{1}{2\sigma^2} \frac{1}{N_{e^{\text{tr}}}} \sum_{i \in V_{e^{\text{tr}}}} \frac{1}{N_{e^{\text{te}}}} \sum_{j \in V_{e^{\text{te}}}^{(i)}} \sum_{c=1}^C \sum_{c' \neq c} |p_j^{\text{ht}}(c'|c) - p_i^{\text{ht}}(c'|c)|$$

$$\tag{120}$$

$\square$

Now we can proceed with the proof of Theorem G.9.

Proof. Directly replace the result of Lemma 5 in Ma et al. [2021] with that of Lemma G.10 and following the proof of Lemma 6 and Theorem 3 in Ma et al. [2021], under Assumption 4.3, we finish the proof of Theorem G.9. $\square$

H Limitations

The theoretical analysis is limited to linear GNNs. However, there some justification for using a linear GNN. 1) Some recent works [Zhu and Koniusz, 2021, Wang et al., 2021b] observed that linear GNNs achieve comparable performance to nonlinear GNNs. Tang and Liu [2023] also theoretically proved that SGC can outperform GCN under some mild assumptions. 2) many recent works on the theoretical analysis of graphs/OOD generalization adopt linear networks [Lin et al., 2023, Wu et al., 2022b, Mao et al., 2023]. 3) Our theory matches the experimental results on the nonlinear GCN and GAT that CIA outperforms IRM and VREx.

We didn't implement some of the node-level OOD methods in our main experiments including Li et al. [2023a], Liu et al. [2023] because they didn't release the code. Another limitation is that we didn't provide an in-depth theoretical comparison between our method and existing node-level OOD methods, but only revealed the difficulty of invariant learning on graphs through the examples of VREx and IRM. However, this theoretical finding is sufficient to inspire the CIA solution and its enhanced version, CIA-LRA. We reserve a more comprehensive analysis of the failure of OOD methods on graphs and broader guidance for designing graph-OOD work for future research.

I Broader Impacts

Our work contributes to improving the node-level OOD generalization of GNN models. We believe our work could positively impact various fields by improving predictive accuracy in areas like healthcare, social networking, etc., potentially leading to better-personalized services and enhanced safety.

48

J Compute Resources

We use one NVIDIA GeForce RTX 3090 or 4090 GPU for each single experiments. All algorithms except EERM and GTrans can be executed on a single 24GB GPU when processing the largest dataset, Arxiv, without encountering out-of-memory.

49

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction concisely state how we dissect the challenges of invariant learning on graphs and out strategies to tackle it. In the introduction, we list all our contributions and indicate the sections in which they are made.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We point out the limitations in Appendix H: we didn't compare the empirical results with some of the node-level OOD works because their codes are unavailable; we didn't theoretically analyze all invariant learning methods.

   Guidelines:

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

3. **Theory Assumptions and Proofs**

50

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

Answer: [Yes]

Justification: All supplementary assumptions, formal versions of the theorems, and full proofs can be found in Appendix G.

Guidelines:

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

4. **Experimental Result Reproducibility**

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

Answer: [Yes]

Justification: We have provided all basic experimental settings and the hyperparameter search space in Appendix C to reproduce all our results. The detailed training procedure of CIA-LRA is provided in Appendix 1. We will release our implementation of CIA and CIA-LRA following the acceptance of our work.

Guidelines:

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

51

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

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

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

   Answer: [No]

   Justification: The datasets utilized in our paper are publicly accessible and remain unmodified. We will release our code following the acceptance of our work, complete with detailed instructions to ensure the faithful reproduction of the main experimental results.

   Guidelines:

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

6. **Experimental Setting/Details**

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

   Answer: [Yes]

   Justification: We adopt the default training details open-source GOOD benchmark, and we have stated some modifications of the setting in the main text and the appendix.

   Guidelines:

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

7. **Experiment Statistical Significance**

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

   Answer: [Yes]

   Justification: The improvements offered by CIA-LRA exceed the error bars of the best existing methods. Also, CIA significantly outperforms VREx and IRM across most dataset splits.

52

Guidelines:

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

8. **Experiments Compute Resources**

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

Answer: [Yes]

Justification: In Appendix J, we report the type of GPU we use and the memory costs.

Guidelines:

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

9. **Code Of Ethics**

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

Answer: [Yes]

Justification: Our research conforms, in every respect, to the NeurIPS Code of Ethics.

Guidelines:

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

10. **Broader Impacts**

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

Answer: [Yes]

53

Justification: We discuss the potential impacts of out work in Appendix I.

Guidelines:

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

11. **Safeguards**

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

Answer: [NA]

Justification: **[TODO]**

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: Yes. All assets used in our research, including code, data, and models, are properly credited.

Guidelines:

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

54

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

13. **New Assets**

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

    Answer: [NA]

    Justification: **[TODO]**

    Guidelines:

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

14. **Crowdsourcing and Research with Human Subjects**

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

    Answer: [NA]

    Justification: **[TODO]**

    Guidelines:

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

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

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

    Answer: [NA]

    Justification: **[TODO]**

    Guidelines:

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

55

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

56