Adversarial Data Augmentations for Out-of-Distribution Generalization

Simon Zhang [1] **Ryan P. DeMilt** [2] **Kun Jin** [2] **Cathy H. Xia** [3]

Abstract

Out-of-distribution (OoD) generalization occurs when representation learning encounters a distribution shift. This occurs frequently in practice when training and testing data come from different environments. Covariate shift is a type of distribution shift that occurs only in the input data, while the concept distribution stays invariant. We propose RIA - Regularization for Invariance with Adversarial training, a new method for OoD generalization under convariate shift, that performs an adversarial search for training data environments. These new environments are induced by adversarial data augmentations that prevent a collapse to an in-distribution trained learner. It works with many existing OoD generalization methods for covariate shift that can be formulated as constrained optimization problems. We develop an alternating gradient descent-ascent algorithm to solve the problem, and perform extensive experiments on OoD graph classification for various kinds of synthetic and natural distribution shifts. We demonstrate that our method can achieve high accuracy compared with OoD baselines.

1. Introduction

The out-of-distribution (OoD) generalization problem is an important topic in machine learning (Li et al., 2022; Shen et al., 2021) where one attempts to extrapolate from training data to in-the-wild distribution shifted data. For example, in computer vision this is commonly demonstrated by the example of identifying cows vs. camels on green or sandy backgrounds (Beery et al., 2018) or the colored MNIST example from (Arjovsky et al., 2019). Covariate shift is when the covariate, or input, distribution shifts while the concept distribution does not change. These varying data conditions are known as varying environments, which can be defined as data distributions conditioned on some varying environmental factors. A covariate shift is an example of a change in environment. Common approaches such as Empirical Risk Minimization (ERM), which selects a model with minimal loss over the training set, cannot generalize to OoD test data as the training environment(s) often rarely reflect the testing environments. Thus OoD generalization requires specialized methods and assumptions beyond minimizing the loss on the training environment.

When there is covariate shift, the distribution of input data shifts due to the change of environments. For various reasons, there may be a scarcity of training environments. It is common, in fact, to just have a few, or possibly one, training environment. Existing OoD generalization methods are based on the concept of achieving invariance, or stability amongst learners on various environments. Due to the lack of diverse training environments, there is a possibility of such a learner collapsing to an ERM solution.

Non-Euclidean data such as graphs offer new challenges to the problem of OoD generalization. The primary challenge is the variable structure of the graphs. The number of nodes of each graph is variable and the interconnection structure of a graph is represented by a 0-1 matrix space different from the graph signal space of node attributes. It is particularly computationally expensive to handle the edges whose count grows quadratically in the number of nodes. Both tensors must be accounted for to define a graph. Furthermore, graphs have the permutation invariance inductive bias.

We will assume a common concept distribution across environments and only covariate shift exists between training and testing distributions. Existing OoD solution methods do not prevent the collapse to an ERM solution during training due to a lack of diverse training environments. We design an algorithm to search, using alternating gradient descent-ascent, for counterfactually generated environments that are hard to learn. This adversarial search prevents collapse to an ERM solution by introducing difficult and diverse environments.

The contributions of this paper are as follows:

[1]Department of Computer Science, Purdue University, West Lafayette, USA [2]Department of Computer Science and Engineering, The Ohio State University, Columbus Ohio, USA [3]Department of Industrial and Systems Engineering, The Ohio State University, Columbus Ohio, USA. Correspondence to: Simon Zhang <zhan4125@purdue.edu>.

Published at the ICML 2023 Workshop on Spurious Correlations, Invariance, and Stability. Honolulu, Hawaii, USA.

- We identify a common issue with many existing OoD solutions, namely when there is a "collapse", or fitting, to the ERM solution.
- We introduce RIA: Regularization for Invariance with Adversarial training, a method to learn more environments for improved OoD generalization. The data exploration process corresponds to adversarially expanding the search space for training environments.
- We perform extensive experiments to demonstrate the effective OoD generalizability of our method on real world as well as synthetic datasets by comparing with existing graph OoD generalization approaches.

2. Related Work

A common approach to tackling the OoD problem is to find a representation that performs stably across multiple environments (Arjovsky et al., 2019; Bagnell, 2005; Ben-Tal et al., 2009; Chang et al., 2020; Duchi et al., 2016; Krueger et al., 2021; Liu et al., 2021a; Mahajan et al., 2021; Mitrovic et al., 2020; Sinha et al., 2017). The goal of such an approach is to eliminate spurious or shortcut correlations that would normally be learned through empirical risk minimization (ERM). ERM is the common approach taken in machine learning to minimize the training error over a union of training environments in order to achieve well known generalization bounds (Vapnik, 1991a). For graph data, (Wu et al., 2022b) assume an underlying data generation process, then their assumptions provide a guarantee (Xie et al., 2020) that they can learn a representation that is stable across environments. In their data generation assumptions, they assume graph data can be decomposed into causal and spurious parts. By learning stably across environments, their objective is to learn to ignore the spurious parts of the data.

Non-Euclidean data such as graphs offer new challenges to the OoD problem. Many of the existing works on this topic are explained in the survey (Li et al., 2022).

3. The Problem and Assumptions

Let a labeled graph, denoted by $(\mathbf{G}, \mathbf{Y})$, be described by the pair of pairs $((\mathbf{X}, \mathbf{A}), \mathbf{Y})$, where the variable $\mathbf{X}$ is the node attribute signal, $\mathbf{A}$ is the 0-1 symmetric adjacency matrix describing the graph structure, and $\mathbf{Y}$ is the ground truth label. The goal is to predict $\mathbf{Y}$ from $\mathbf{G} = (\mathbf{X}, \mathbf{A})$, where the covariate distribution $P_e(\mathbf{X}, \mathbf{A})$ depends on the environment e, and the concept distribution $P(\mathbf{Y}|(\mathbf{X}, \mathbf{A}))$ does not change.

Definition 3.1. Denote $\mathcal{E}_{all}$ the set of all environment indices that index all data distributions for some classification task that we want to learn. Let $\mathcal{E}_{tr} \subset \mathcal{E}_{all}$ be a set of training environments that are accessible during training.

We assume that there is a shift in the covariate distribution for testing different from the training distribution. The out-of-distribution generalization problem seeks to predict a graph label on any unseen testing distribution. Since we do not know the testing distribution(s), we optimize for the worst case data distribution in the following minimax optimization problem.

$$\min_{h} \sup_{e \in \mathcal{E}_{all}} \mathbb{E}_{(\mathbf{G}^e, \mathbf{Y}^e) \sim P_e} [l_e(h(\mathbf{G}^e), \mathbf{Y}^e)] \tag{1}$$

where e indexes a specific environment, P_e is the distribution from which input data is drawn, and $h(\cdot)$ is a classifier to predict ground truth label $\mathbf{Y}$. We assume there is a convex loss for each environment, called l_e, for $e \in \mathcal{E}_{all}$ (Arjovsky et al., 2019). The expected loss over an environment is oftentimes called the risk. The environmental risk is denoted by the symbol R_e.

For the data generation, we assume there are causal and spurious random variables $\mathbf{X}_C, \mathbf{X}_S$ representing the graph signal as well as causal and spurious variables $\mathbf{A}_C, \mathbf{A}_S$ representing the graph connectivity. The causal and spurious signals and connectivity join together to induce the input graph $\mathbf{G} = (\mathbf{X} = J_X(\mathbf{X}_C, \mathbf{X}_S), \mathbf{A} = J_A(\mathbf{A}_C, \mathbf{A}_S)$. Let the graph $\mathbf{C} = (\mathbf{X}_C, \mathbf{A}_C)$ be the pairing of the causal signal and connectivity. The ground truth label $\mathbf{Y}$ is generated by the following composition: $\mathbf{Y} = m(\mathbf{C}, \eta)$ for η a random exogenous variable. The graph $\mathbf{C}$ is called a causal graph. The causal graph $\mathbf{C}$ satisfies $\mathbf{Y} \perp\!\!\!\perp (\mathbf{X}, \mathbf{A})|\mathbf{C}$ and determines the label $\mathbf{Y}$ up to an η. The map m is by a graph data generation process due to some Structural Causal Model (see Definition A.1 in Appendix for more details). Many OoD generalization methods seek to find the graph $\mathbf{C}$ from the graph $\mathbf{G}$.

ERM: When there is no distribution shift at all, the standard approach would be to take $\mathcal{E}_{tr}$, and minimize the average risk over these training environments. This is known as Empirical Risk Minimization (ERM), which is given in the following equation:

$$\min_{h} \frac{1}{|\mathcal{E}_{tr}|} \sum_{e \in \mathcal{E}_{tr}} \mathbb{E}[l_e(h(\mathbf{G}^e), \mathbf{Y}^e)] \tag{2}$$

However, this does not generalize to when there is a distribution shift of $P(\mathbf{G}^e)$ from training environments with $e \in \mathcal{E}_{tr}$ to testing distributions with $e \in \mathcal{E}_{all}$ (Ahuja et al., 2021).

3.1. Failing to Extrapolate like ERM: ERM Collapse

ERM Collapse refers to when an OoD generalization method behaves similarly to ERM on both training and testing data. Therefore, ERM collapse denotes when an OoD method fails to extrapolate to OoD data because it treats the OoD data the same as in-distribution data. This results in a mimicking of the training and testing behavior of ERM. This can

occur when the training environments are all very similar to each other. In this case, when an OoD generalization method is formulated as a constrained optimization method, the constraint becomes ineffective, and the optimization problem collapses to ERM. Stabilizing only two similar training environments, for example, may be insignificant as an optimization constraint. In some OoD generalization methods, certain causal assumptions beyond those given in Section 3 and Section A are assumed. For example, a deterministic map from the data to the causal part may be assumed. This assumption may not hold in the data generation process, e.g. the Motif dataset from (Wu et al., 2022a). In Motif, the graph data are generated by attaching a causal motif and a spurious graph. There is no guarantee that there can be a deterministic map from the graph data to its causal subgraph. The map may not be well defined since there could be many such subgraphs in a given graph. Some formal explanations for when an OoD generalization method can become "spurious free," or independent of spurious attributes, are given in (Chen et al., 2022). Formally speaking, ERM collapse implies not being "spurious free." ERM collapse also implies many other characterizations of failures to extrapolate OoD.

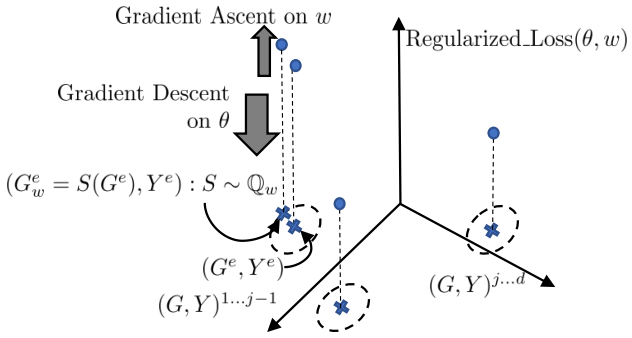

Figure 1. Geometric view of the minimax optimization procedure RIA algorithm on Regularized_Loss(θ, w) as given in Equation (5) where w indexes the artificial search environments, θ indexes the learner's neural weights.

4. Method

We design a training algorithm for OoD generalization that adversarially explores data points by data augmentation for extrapolation beyond the training environments for OoD generalization. We focus on graph data, however our method can be generalized to any kind of data. The exploration is done by stochastic gradient ascent updates, adversarially maximizing against the ERM loss of any regularized OoD loss to search over environments (Yi et al., 2021). By learning a distribution of data augmentations and not a single data augmentation, we ensure diversity of solutions in addition to OoD robustness (Wang et al., 2021a).

There are many existing OoD generalization methods and architectures. It is common for these generalization methods to be formulated as a constrained optimization equation with ERM as the minimization objective. A constrained optimization equation such as

$$\begin{aligned} &ERM(h) \\ &\text{s.t. } C(h) \end{aligned} \tag{3}$$

can be reformulated as regularized ERM loss of the form $ERM(h)$ + **OoD-Reg.**(h) with $ERM(h)$ from Equation (2). The minimization of **OoD-Reg.**(h) to zero is a sufficient condition to obtaining the constraint $C(h)$. The constraint $C(h)$ usually imposes invariance to spurious correlations from training data.

Some common constrained optimization methods that are reformulated as regularized ERM losses (2) to form an OoD generalization loss include IRM (Arjovsky et al., 2019), VREx (Krueger et al., 2021), and RICE (Wang et al., 2022). IRM is a constrained optimization method that learns a representation whose correlation between the representation and the label across multiple training environments is invariant. VREx is also a constrained optimization method that guarantees that the variance across environmental risks is low. RICE studies OoD generalization with causal invariant transformations. It shows that if such transformations are available, then one can learn a minimax optimal model across the domains using only single domain data. It proposes a regularized training procedure for OoD generalization on a combination of the training environments. For more information about the three OoD generalization methods as constrained optimization problems and their implementation, see Section B in Appendix. We introduce adversarial data augmentations to search for a robust OoD solution preventing collapse to the ERM solution. Our data generation assumptions, as given in the Appendix, are compatible with all three methods.

We introduce a distribution of data augmentations and aim to find an approximate distribution that maximizes the ERM loss, preventing a collapse to the ERM solution.

Definition 4.1. Let $\mathbb{Q}_w$ be a distribution indexed by w of data augmentations on graphs, with each augmentation denoted as $\mathbf{S}$, so that for a given classifier h and a set of training environments $\mathcal{E}_{tr}$,

$$\begin{aligned} &\mathbb{Q}_{max}(h, \{(\mathbf{G}^e, \mathbf{Y}^e)\}_{e\in\mathcal{E}_{tr}}) = \\ &\text{argmax}_{\mathbb{Q}_w} \frac{1}{|\mathcal{E}_{tr}|} \sum_{e\in\mathcal{E}_{tr}} \mathbb{E}_{\mathbf{S}\sim\mathbb{Q}_w}[l_e(h, \mathbf{S}(\mathbf{G}^e), \mathbf{Y}^e)] \end{aligned} \tag{4}$$

The purpose of the argmax in Definition 4.1 is to skew the distribution on the pushforward distributions $(\mathbf{S}')_{\#}(P_{tr}) := P_{tr} \circ \mathbf{S}'^{-1}, \mathbf{S}' \sim \mathbb{Q}_{max}$ towards the hardest data augmentations.

Dataset (acc)	CMNIST ↑		SST2↑		Motif ↑				AMotif↑				Synth ↑	
covariate	color		length		basis		size		basis		size		basis+std, $r = 1$	
	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD
RIA-RICE	61.7±1.6	48.1 ± 0.8	89.4±0.6	81.9 ± 0.2	92.4 ± 0.2	65.1±5.9	92.4 ± 0.2	55.3 ± 0.4	79.3±1.6	36.8±4.2	67.4±1.5	33.4±1.3	48.0±9.0	58.5±1.5
RIA-IRM	65.5 ± 2.8	41.6 ± 0.6	89.7 ± 0.6	81.7 ± 0.5	33.7 ± 0.8	33.9 ± 0.7	33.5 ± 0.8	34 ± 2.9	89.6 ± 0.8	40.5 ± 3.8	48.6 ± 0.4	48.6 ± 2	51 ± 0.6	54 ± 0.8
RIA-VREx	79.3 ± 0.7	38.7 ± 0.7	89.8 ± 2	80.2 ± 4	32.2 ± 2.3	34 ± 1.7	33.5 ± 0.5	34 ± 1.0	90.5 ± 4.5	42.4 ± 0.6	90.3 ± 0.9	47 ± 0.87	40 ± 0.8	60 ± 1.9
ERM	77.5±0.5	28.3±0.3	89.4±0.4	81.2±0.2	92.3±0.3	68.3±0.3	92.1±0.1	51.4±0.4	80.8±1.1	33.2±1.0	67.9±2.2	33.2±1.0	53.5±1.5	53.5±1.5
DIR	39 ± 2.9	28.1 ± 10	83.6 ± 4.6	81.1 ± 4.9	82.2 ± 5.2	73.6 ± 5.8	75.6 ± 3.9	39.3 ± 1	34.7 ± 2.5	35 ± 2.9	36.3 ± 5.2	33.1 ± 3.3	48 ± 1.2	61 ± 1.4
RICE	68.2±0.9	26.3±0.5	90.0 ± 0.2	80.7±0.7	92.4±0.2	65.1±5.9	92.2±0.0	55.1±0.2	69.3±9.8	36.2±1.7	50.5±9.2	33.5±1.2	54.5±2.5	54.0±1.0
Coral	78.3 ± 0.3	29.0±0.0	89.3±0.3	79.4±0.4	92.3±0.3	68.4±0.4	92.1±0.1	50.5±0.5	81.0±0.2	33.9±1.3	67.9±0.6	32.9±0.8	54.0±2.0	51.5±2.5
DANN	77.5±0.5	29.1±0.6	89.3±0.8	79.4±0.9	92.3±0.8	65.2±0.7	92.1±0.6	51.2±0.7	81.1±0.2	38.1±1.4	69.2±1.1	33.1±0.5	54.5±1.8	52.0±0.5
GroupDRO	77.0±1.0	28.5±0.5	88.8±0.8	80.7±0.7	91.8±0.8	67.6±0.6	91.6±0.6	51.0±1.0	74.0±1.0	38.6±0.6	83.9±0.8	35.8±0.8	50.5±0.5	52.5±0.5
GSAT	67.0±2.6	39.9±0.6	89.0±0.1	80.6±1.1	92.5 ± 0.0	57.1±6.8	92.1±0.1	53.3±0.3	69.3±9.8	36.2±1.7	50.5±9.2	33.5±1.2	58.5±7.5	50.5±6.5
IRM	77.0±1.0	26.9±0.9	88.7±0.7	79.0±1.0	91.8±0.8	69.8±0.8	91.6±0.6	50.9±0.9	79.0±1.0	37.9±0.9	79.6±0.6	33.6±0.6	62.5 ± 0.5	48.5±0.5
Mixup	76.7±0.7	25.7±0.7	88.9±0.9	79.9±0.9	91.8±0.8	69.5±0.5	91.5±0.5	50.7±0.7	70.9±0.9	36.7±0.7	68.7±0.7	33.0±1.0	41.5±0.5	58.5±0.5
VREx	77.0±1.0	27.7±0.7	88.8±0.8	79.8±0.8	91.8±0.8	70.7 ± 0.7	91.6±0.6	51.8±0.8	78.6±0.6	33.9±0.9	65.6±0.6	34.0±1.0	50.5±0.5	52.5±0.5
DropEdge	56.9±0.9	19.7±0.7	88.8±0.8	81.7±0.7	34.7±0.7	31.5±0.5	34.8±0.8	31.6±0.6	37.9±0.9	33.9±0.9	33.8±0.8	33.0±1.0	59.5 ± 0.5	43.5±0.5

Table 1. Accuracy of all baseline approaches as well as RIA-RICE, RIA-IRM, RIA-VREx on all datasets under different covariate shifts. For each covariate shift, the columns labeled ID refer to the in-distribution test accuracies while the columns labeled OOD refer to the out-of-distribution test scores. Red and gray entries are the max and second max test accuracies, respectively, for each column.

Hardest refers to data augmentations that are farthest from allowing collapse to an ERM solution.

Our minimax optimization problem can then be formulated as follows:

$$\min_h \frac{1}{|\mathcal{E}_{tr}|} \sum_{e \in \mathcal{E}_{tr}} \mathbb{E}_{P_{tr}^{Aug}(h)}[\textbf{OoD-Reg}_\bullet(h((\mathbf{G}^e)'), \mathbf{Y}) + l_e(h, (\mathbf{G}^e)', \mathbf{Y}^e)] \tag{5}$$

where $P_{tr}^{Aug}(h) = P[(\mathbf{G}^e)' = \mathbf{S}(\mathbf{G}^e), \mathbf{Y}^e)]$ satisfies $\mathbf{S} \sim \mathbb{Q}_{max}(h, \{(\mathbf{G}^e, \mathbf{Y}^e)\}_{e \in \mathcal{E}_{tr}})$.

The minimization of the regularization $\textbf{OoD-Reg}_\bullet(h)$ provided by existing OoD generalization methods allows for stabilization across environments and extrapolation to an OoD test dataset. This cannot occur if there is ERM collapse. The adversarially trained data augmentations help push the data away from ERM collapse. Intuitively, equation (5) aims to find the optimal OoD generalization classifier that minimizes the worst-case ERM loss, achieved via data augmentation. See Appendix Figure 3 for how this loss behaves during training and testing.

4.1. Algorithm

To solve the minimax optimization equation posed in Equation (5), we propose an alternating gradient descent-ascent algorithm, which is shown in Algorithm 1. In the algorithm, the GNN f_w, with neural weights w, determines a tensor of Bernoulli probabilities for which an adversarial data augmentation with k entries is sampled. The GNN h_θ is some graph representation learner.

A geometric view of the optimization algorithm is shown in Figure 1. In our implementation, we learn a distribution of node attribute masking data augmentations to prevent ERM collapse.

Algorithm 1 RIA by Alternating SGD with Adversarial Data Augmentation for OoD Generalization on Graphs

Require: Training graph data $(G_i^e = (X_i^e, A_i^e), Y_i^e)$, $G_i^e \in P_{n_e}^e \sim (P^e)^{n_e}$, $e \in \mathcal{E}_{tr}, i = 1..n_e$; n_e the number of training data for environment e. Parameters of minimizing/maximizing GNN: θ/w, Learning rates lr_θ, lr_w, k: Number of entries of X_i^e to keep, $\textbf{OoD-Reg}_\bullet$ is an OoD generalization regularizer from some existing method. T is the ratio of num. maximization to num. minimization steps
while not converged or max epochs not reached **do**
  for $t = 1...T$ **do**
    for $e = 1...|\mathcal{E}_{tr}|$ **do**
      $M_w^{e,i} \leftarrow s(\sigma((f_w(X_i^e, A_i^e)))$; for $i = 1..n_e$ // f_w is a GNN; s is a 0-1 sampler from a tensor of Bernoulli probs., sampling k times to update a tensor of 0's.
      $G_w^{e,i} \leftarrow (M_w^{e,i} \odot X_i^e, A_i^e)$
    end for
    $E(w, \theta) \leftarrow \frac{1}{|\mathcal{E}_{tr}|} \sum_{e=1}^{|\mathcal{E}_{tr}|} \frac{1}{n_e} \sum_{i=1}^{n_e} [l_e(h_\theta, G_w^{e,i}, Y_i^e)]$
    $J(w, \theta) \leftarrow \frac{1}{|\mathcal{E}_{tr}|} \sum_{e=1}^{|\mathcal{E}_{tr}|} \frac{1}{n_e} \sum_{i=1}^{n_e} [\textbf{OoD-Reg}_\bullet(h_\theta, G_w^{e,i}, Y_i^e)] + E(w, \theta)$
    Update $w \leftarrow w + lr_w \cdot \nabla_w E(w, \theta)$
    if t==T **then**
      Update $\theta \leftarrow \theta - lr_\theta \cdot \nabla_\theta J(w, \theta)$;
    end if
  end for
end while

5. Experiments

We ran all our experiments on a 64 core Intel(R) Xeon(R) CPUs @2.40 GHz with 128 GB DRAM equipped with one 40 GB DRAM Ampere A100 GPU. The corresponding test scores for the best in-distribution validation score are averaged across 3 runs for both real world and synthetic datasets. Hyperparameters follow the defaults of the GOOD

benchmark (Gui et al., 2022), see the Appendix.

We implement Algorithm 1 (referred to as RIA in Table 1) using the regularizations of RICE, IRM, VREx. We compare our approach with the baselines of Coral (Sun & Saenko, 2016), DANN (Ganin et al., 2016), DIR (Wu et al., 2022b), ERM (Vapnik, 1999), GSAT (Miao et al., 2022), GroupDRO (Sagawa et al., 2019), IRM (Arjovsky et al., 2019), Mixup (Wang et al., 2021b), RICE (Wang et al., 2022), VREx (Krueger et al., 2021), EdgeDrop (Rong et al., 2020) all implemented in the GOOD (Gui et al., 2022) benchmark.

Additive Spurious Attributes Synthetic Dataset: We develop a synthetic binary classification dataset that models a noisy data generation process as in the SCM in Appendix Figure 2. For more information on the dataset, see Appendix, section C. It is designed to model attribute shifts instead of just shifts in the graph topologies as in Motif.

Real World Graph Classification Experiments: We also perform experiments on real world benchmarks. For all the scores, see Table 1. We use the datasets of CMNIST (Arjovsky et al., 2019), SST2 (Liu et al., 2021b), and Motif (Wu et al., 2022b) from the GOOD framework as well as AMotif, a modification of Motif. Each of these datasets follows the causal model as shown in Appendix Figure 2. Accuracy is used to measure the performance on all the datasets, as is standard. Each dataset involves different kinds of covariate shift. For more details about each dataset and the kind of covariate shift imposed on them, see the Appendix.

As shown in Table 1, our method, RIA, performs well both in the in-distribution ID and out-of-distribution OoD settings. For the ID case, RIA performs the highest or second highest on all datasets in at least one method except for the synthetic dataset. This suggests that even in the ID setting, the data is never truly in-distribution. There is always some benefit to pushing away from the ERM solution. For the OoD case, the adversarial data augmentations seem able to counterfactually generate environments similar to the testing input data. This is the benefit to minimax optimization. Of course there is no guarantee that RIA is converting the training distribution into the testing distribution exactly. However, the training distribution is no longer the same thing. RIA obtains the highest or second highest score for every dataset except Motif by at least one method. The performance on Motif is not high since Motif has very simple attributes. The ablation comparison between each existing method: IRM, RICE, VREx, and RIA applied to it are included in Table 1. We see that RIA not only improves upon the existing method, but oftentimes outperforms many other baselines.

6. Discussion

We observe widespread ERM collapse in existing methods in our experiments. Many of the methods such as IRM, VREx, Mixup and DropEdge behave very similar to ERM. We believe that these particular methods do not veer from ERM aggressively enough. IRM and VREx, may not have enough training environments. Mixup and DropEdge, as static data augmentations, are not actually changing the training distribution or achieving any kind of invariance across environments. RIA prevents ERM collapse and due to the adversarial generation of environments against the ERM loss the learner has enhanced robustness.

Although we only did experiments on graph data, we believe RIA can easily be implemented for images and other data modalities. One caveat we have observed empirically is that the data augmentations should be diverse and only slightly affect the training distribution. Sudden changes to the training distribution can over-correct the learner.

7. Conclusion

We have introduced adversarial data augmentations to provide a search for a robust OoD solution. We formulate and motivate the OoD problem as a minimax optimization problem over a set of environments. To address the lack of training environments and to prevent an early collapse of the classifier onto an ERM solution on the training distribution during OoD training, we propose RIA: Regularization for invariance with adversarial training. We compare our approach, RIA, with state of the art OoD generalization approaches including DIR(Wu et al., 2022b) and RICE (Wang et al., 2022) as well as the classical ERM on graphs. This shows that for graph classification, preventing ERM collapse in the OoD setting improves existing OoD generalization methods.

Acknowledgment

This work was supported in part by the National Science Foundation under Grant OAC-2310510.

References

Ahuja, K., Caballero, E., Zhang, D., Gagnon-Audet, J.-C., Bengio, Y., Mitliagkas, I., and Rish, I. Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 34:3438–3450, 2021.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Bagnell, J. A. Robust supervised learning. In *AAAI*, pp. 714–719, 2005.

Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473, 2018.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. Robust optimization. In *Robust optimization*. Princeton university press, 2009.

Chang, S., Zhang, Y., Yu, M., and Jaakkola, T. Invariant rationalization. In *International Conference on Machine Learning*, pp. 1448–1458. PMLR, 2020.

Chen, Y., Xiong, R., Ma, Z.-M., and Lan, Y. When does group invariant learning survive spurious correlations? *Advances in Neural Information Processing Systems*, 35: 7038–7051, 2022.

Dey, T. K. and Zhang, S. Approximating 1-wasserstein distance between persistence diagrams by graph sparsification. *arXiv preprint arXiv:2110.14734*, 2021.

Donini, M., Oneto, L., Ben-David, S., Shawe-Taylor, J. S., and Pontil, M. Empirical risk minimization under fairness constraints. *Advances in neural information processing systems*, 31, 2018.

Duchi, J., Glynn, P., and Namkoong, H. Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv preprint arXiv:1610.03425*, 2016.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Gui, S., Li, X., Wang, L., and Ji, S. Good: A graph out-of-distribution benchmark. *arXiv preprint arXiv:2206.08452*, 2022.

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Le Priol, R., and Courville, A. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

Li, H., Wang, X., Zhang, Z., and Zhu, W. Out-of-distribution generalization on graphs: A survey. *arXiv preprint arXiv:2202.07987*, 2022.

Liu, J., Hu, Z., Cui, P., Li, B., and Shen, Z. Heterogeneous risk minimization. In *International Conference on Machine Learning*, pp. 6804–6814. PMLR, 2021a.

Liu, M., Luo, Y., Wang, L., Xie, Y., Yuan, H., Gui, S., Yu, H., Xu, Z., Zhang, J., Liu, Y., Yan, K., Liu, H., Fu, C., Oztekin, B. M., Zhang, X., and Ji, S. DIG: A turnkey library for diving into graph deep learning research. *Journal of Machine Learning Research*, 22 (240):1–9, 2021b. URL http://jmlr.org/papers/v22/21-0343.html.

Mahajan, D., Tople, S., and Sharma, A. Domain generalization using causal matching. In *International Conference on Machine Learning*, pp. 7313–7324. PMLR, 2021.

Miao, S., Liu, M., and Li, P. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pp. 15524–15543. PMLR, 2022.

Mitrovic, J., McWilliams, B., Walker, J. C., Buesing, L. H., and Blundell, C. Representation learning via invariant causal mechanisms. In *International Conference on Learning Representations*, 2020.

Rong, Y., Huang, W., Xu, T., and Huang, J. Dropedge: Towards deep graph convolutional networks on node classification. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=Hkx1qkrKPr.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

Shen, Z., Liu, J., He, Y., Zhang, X., Xu, R., Yu, H., and Cui, P. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Sinha, A., Namkoong, H., Volpi, R., and Duchi, J. Certifying some distributional robustness with principled adversarial training. *arXiv preprint arXiv:1710.10571*, 2017.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Sun, B. and Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV*

2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14, pp. 443–450. Springer, 2016.

Vapnik, V. Principles of risk minimization for learning theory. In Moody, J., Hanson, S., and Lippmann, R. (eds.), *Advances in Neural Information Processing Systems*, volume 4. Morgan-Kaufmann, 1991a. URL https://proceedings.neurips.cc/paper/1991/file/ff4d5fbbafdf976cfdc032e3bde78de5-Paper.pdf.

Vapnik, V. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991b.

Vapnik, V. N. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

Wang, H., Xiao, C., Kossaifi, J., Yu, Z., Anandkumar, A., and Wang, Z. Augmax: Adversarial composition of random augmentations for robust training. *Advances in neural information processing systems*, 34:237–250, 2021a.

Wang, R., Yi, M., Chen, Z., and Zhu, S. Out-of-distribution generalization with causal invariant transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 375–385, 2022.

Wang, Y., Wang, W., Liang, Y., Cai, Y., and Hooi, B. Mixup for node and graph classification. In *Proceedings of the Web Conference 2021*, pp. 3663–3674, 2021b.

Wood, J. and Shawe-Taylor, J. Representation theory and invariant neural networks. *Discrete applied mathematics*, 69(1-2):33–60, 1996.

Wu, Q., Zhang, H., Yan, J., and Wipf, D. Handling distribution shifts on graphs: An invariance perspective. *arXiv preprint arXiv:2202.02466*, 2022a.

Wu, Y.-X., Wang, X., Zhang, A., He, X., and Chua, T.-S. Discovering invariant rationales for graph neural networks. *arXiv preprint arXiv:2201.12872*, 2022b.

Xie, C., Ye, H., Chen, F., Liu, Y., Sun, R., and Li, Z. Risk variance penalization. *arXiv preprint arXiv:2006.07544*, 2020.

Yi, M., Hou, L., Sun, J., Shang, L., Jiang, X., Liu, Q., and Ma, Z. Improved ood generalization via adversarial training and pretraing. In *International Conference on Machine Learning*, pp. 11987–11997. PMLR, 2021.

Zhang, S., Xiao, M., and Wang, H. Gpu-accelerated computation of vietoris-rips persistence barcodes. *arXiv preprint arXiv:2003.07989*, 2020.

Zhang, S., Mukherjee, S., and Dey, T. K. Gefl: Extended filtration learning for graph classification. In *Learning on Graphs Conference*, pp. 16–1. PMLR, 2022.

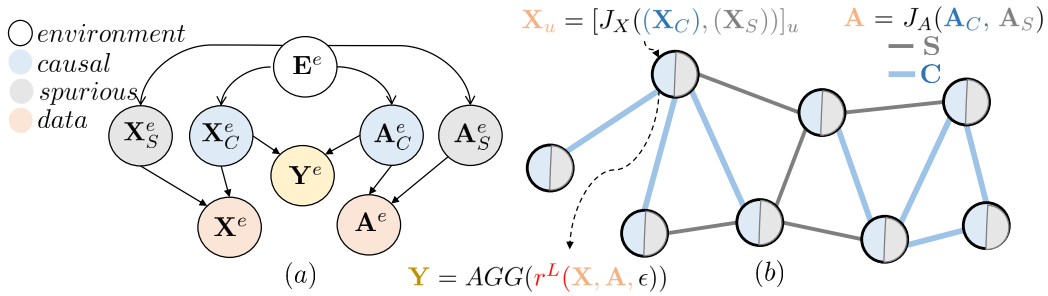

Figure 2. (a): A common casual graph for the data generation process. The variable $\mathbf{E}^e$ determines the number of nodes for environment e. (b) A labeled attributed graph instance with the joining operation for causal/spurious attributes and edges shown. In the figure, the joining operation J_X is shown as the concatenation of causal and spurious attribute tensors. The joining operation J_A shown in the figure provides the sum of A_C and A_S where $A_C \odot A_S = 0$. The half grey color on nodes represents the X_S while the half blue color represents the X_C.

A. A Causal Model for Graph Data

In the formulation of our approach, there is no dependency on a particular OoD loss besides that it can be decomposed into the form $ERM(h)$ + **OoD-Reg.**(h) on learner h. Many existing methods require causal assumptions on the data generation process. We discuss here a particular causal model that satisfies many of these existing data generation processes. It is based on the causal model presented in (Arjovsky et al., 2019). RIA can address when these causal assumptions fail by adversarially augmenting the data so that there are diverse counterfactually generated OoD environments. To actually extrapolate to OoD data *as intended* by existing methods, RIA would have to learn a distribution shift that can push the data towards these causal assumptions. Empirically, we find that preventing ERM collapse allows for OoD generalization even without a guarantee that the intended causal assumptions are satisfied.

Definition A.1. For every environment index e define the following random variables:

1. Let there be an environment random variable $\mathbf{E}^e$ which determines the causal and spurious graph random variables $\mathbf{X}_C^e, \mathbf{X}_S^e, \mathbf{A}_C^e, \mathbf{A}_S^e$

2. $\mathbf{A}^e = J_A(\mathbf{A}_C^e, \mathbf{A}_S^e); \mathbf{X}^e = J_X(\mathbf{X}_S^e, \mathbf{X}_C^e)$ as in Figure 2 where J_X and J_A are called the joining maps. They are permutation equivariant maps on each component, respectively,
This means: $\forall \pi \in Sym([n])\ J_A(\pi \cdot \mathbf{A}_C^e, \pi \cdot \mathbf{A}_S^e) = \pi \cdot \mathbf{A}^e$ and $J_X(\pi \cdot \mathbf{X}_C^e, \pi \cdot \mathbf{X}_S^e) = \pi \cdot \mathbf{X}^e$ where $\pi \cdot A$, for A an $n \times n$ matrix, is the map $A \mapsto PAP^T$ for P the matrix representation of $\pi \in Sym([n])$ and $\pi \cdot X$ is the map $X \mapsto PX$ for X a $n \times d$ matrix
3. $\mathbf{H}^e = r^L(\mathbf{H}^e = (\mathbf{X}_C^e, \mathbf{A}_C^e), \epsilon)$ where r is some recursive message passing function, see Equation (6) on the causal subgraph and ϵ is a random noise variable with $\mathbf{H}^e \perp\!\!\!\perp \epsilon$.
4. $\mathbf{Y}^e = AGG(\mathbf{H}^e)$, AGG is permutation invariant, meaning $\forall \pi \in Sym([n])\ \mathbf{Y}^e = AGG(\pi \cdot \mathbf{H}^e)$
Let $P^e = P(\mathbf{G}^e = (\mathbf{X}^e, \mathbf{A}^e), \mathbf{Y}^e)$ satisfy 1 and 2 above, it is called the data environment distribution.

Definition A.2. $\mathcal{P} = \{P^e | P^e$ satisfies Definition A.1$\}$

Examples of message passing functions on graphs include GNN-like recursive functions such as:

$$X_u^0 = X_{C,u}^e \text{ with } r(X_u^l, A_C) = \Gamma(\{X_u^{l-1} + \epsilon, X_v^{l-1} | v \in Nbr_{A_C}(u)\}) \text{ for } l = 1...L \tag{6}$$

Γ is a permutation invariant function on a set and ϵ is the random noise variable and

$Nbr_{A_C}(u)$ is the set of neighbors of u using edges of A_C

An example of a map m could be a set representation map on the set of attributes of $\mathbf{H}^e$

We further assume that there is a map c_* from $(\mathbf{X}, \mathbf{A})$ to $(\mathbf{X}_C, \mathbf{A}_C)$ for the SCM of Definition A.1.

A.1. Illustrating ERM Collapse

In Figure 3, we show the training and OoD testing losses across 150 epochs of training for ERM, IRM and VREx as well as RIA applied to IRM and VREx. We can see the ERM collapse phenomenon. SST2 does not have as much of a distribution shift so it is harder to observe ERM collapse. CMNIST has a synthetic distribution shift attached to a natural data distribution and only two very similar training environments so it is easier to observe ERM collapse. On CMNIST, VREx and IRM both follow the training loss curve of ERM since they must converge to zero training loss. RIA-VREx and RIA-IRM, on the other hand, are prevented from converging to zero loss. For OoD generalization for both SST2 and CMNIST, we see that by preventing ERM collapse, we can in fact maintain low OoD loss and prevent mimicking the behavior of ERM. The other methods, IRM and VREx, on the other hand, diverge like ERM.

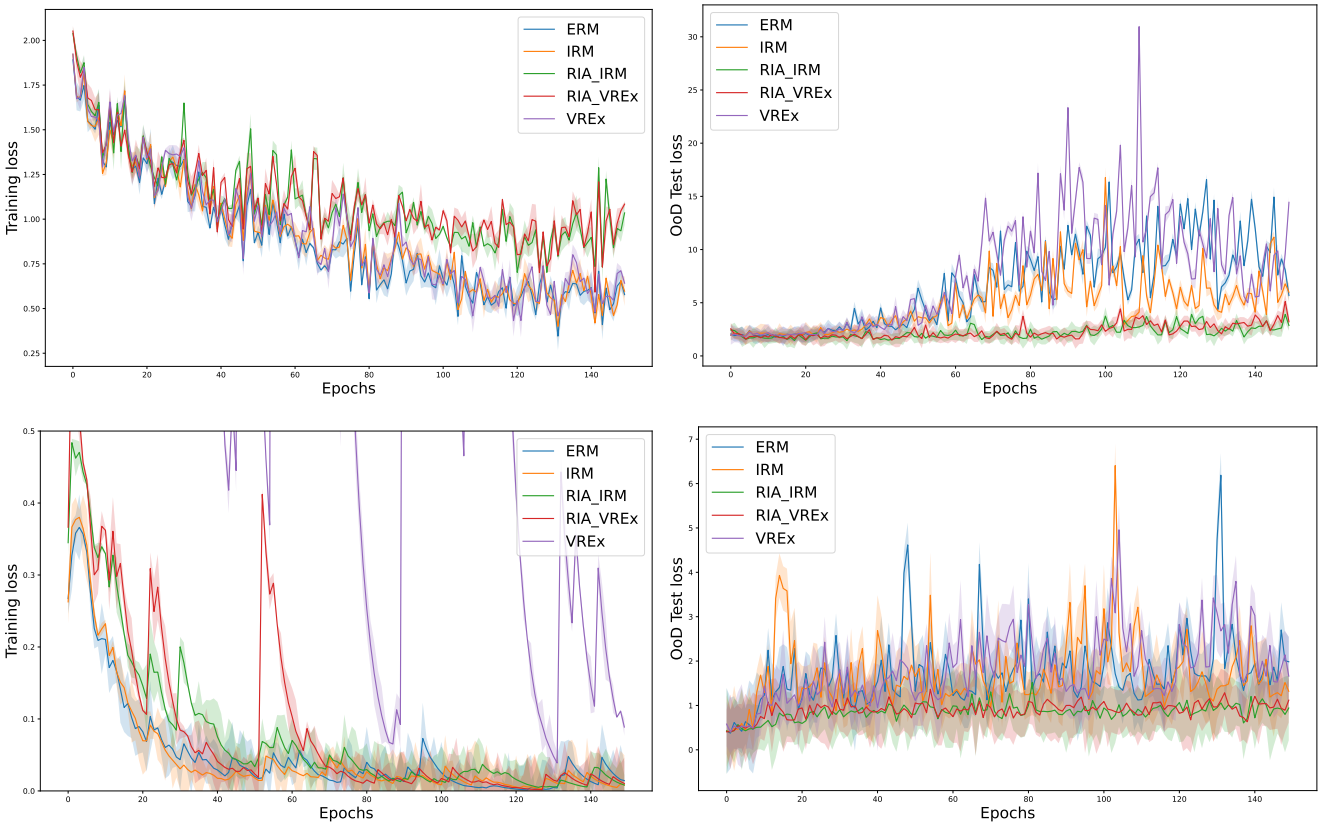

Figure 3. Illustration of ERM Collapse on the CMNIST (above) and SST2 (below) dataset. Left: Training loss where ERM collapse is happening to traditional constrained optimization OoD generalization methods. Red and Green are RIA on IRM and VRex, respectively. Right: Test OoD loss. The consequences of ERM collapse are prevented.

B. Constrained Optimization Problems

Many common problems in machine learning can be formulated as constrained optimization problems. For example, the optimal transport problem can be formulated as:

$$\begin{aligned} &\min_{\gamma\in\Gamma(\mu,\nu)} [\mathbb{E}_{(x,y)\in\gamma} d(x,y)] \\ \text{s.t. } &\int_{X\times X} \gamma(x,y)\cdot dy = \mu(x) \\ &\int_{X\times X} \gamma(x,y)\cdot dx = \nu(y) \\ &\gamma \geq 0 \end{aligned} \tag{7}$$

In (Dey & Zhang, 2021) the Wasserstein distance is defined on $X = \mathbb{S}^2 \cup \{z\}$, the pointed sphere. This requires further constraints for the metric on the sphere. Pointed spaces as constraints have also been used for "coning" a graph (Zhang et al., 2022). Coning means that: 1. every node in a graph is connected by an edge to a special node z and 2. all 3-cycles containing z are filled with a triangle. Adding these constraints affects a graph's homology, viewed as a simplicial complex. When learning the construction of the graph node by node, this can in fact improve the expressivity of graph representation learning. Graphs or spheres are all examples of finite metric space data. Persistent homology is a way to measure the change in homology of a simplicial complex constructed on the data. In computing persistent homology on finite metric space data, it is common to place constraints on how the simplicial complex is constructed from the data. One very natural and computationally efficient (Zhang et al., 2020) constraint is to assume a k-clique, when constructed, must be filled in by a $k+1$ simplex during construction. This constraint on the construction of the simplicial complex makes the simplicial complex a Vietoris-Rips complex.

For deep learning, in general, ERM (Vapnik, 1991b), a well known optimization method that forms the backbone of deep learning. Constraints have been imposed on ERM to induce fairness (Donini et al., 2018) and symmetry invariance, such as in this early work (Wood & Shawe-Taylor, 1996), amongst many things.

When the constraints imposed on a constrained optimization problem are not effective such as in ERM collapse, then the optimization problem becomes am optimization problem over the training data which is agnostic to distribution shift. This results in learning spurious correlations from the training data.

B.1. Invariance in OoD Generalization as Constrained Optimization

We have identified three OoD generalization methods that are formulated as constrained optimization problems: IRM, VREx, and RICE. We go over each method and how they can be rewritten as regularized ERM methods. Regularized ERM methods risk the possiblity of ERM collapse since their constraints may fail to be effective.

Let R_e denote the risk function over a given environment e.

IRM: IRM is the following optimization problem:

$$\begin{aligned} &\min_{\Phi:X\to H, w:H\to Y} \sum_{e\in\mathcal{E}_{tr}} R_e(w\circ\Phi) \\ \text{s.t. } &w \in argmin_{w:H\to Y} R_e(w\circ\Phi), \forall e\in\mathcal{E}_{tr} \end{aligned} \tag{8}$$

This can be written as the following regularized ERM problem called IRMv1 whose minimization implies the IRM constrained optimization problem:

$$\min_{\Phi:X\to Y} \sum_{e\in\mathcal{E}_{tr}} R_e(\Phi) + \lambda\cdot|\nabla w|_{w=1.0} R_e(w\cdot\Phi)|_2 \tag{9}$$

For graph learning, the map Φ can be implemented as a graph representation learner such as a GNN. The w learnable scalar parameter just multiplies the representation before taking the cross entropy loss.

One can check that the causal model of Section A is still compatible with IRM.

VREx: VREx is the following optimization problem:

$$R_{MM-REx}(h) = \max_{\sum_{e\in\mathcal{E}_{tr}} \lambda_e=1, \lambda_e\geq\lambda_{min}} \sum_{e\in\mathcal{E}_{tr}} \lambda_e \cdot R_e(h) = \\ (1 - m \cdot \lambda_{min}) \cdot \max_e R_e(h) + \lambda_{min} \cdot \sum_{e\in\mathcal{E}_{tr}} R_e(h) \tag{10}$$

This can be approximated as the following regularized ERM problem called VREx whose minimization gives a smoother version of the MM-REx constrained optimization problem:

$$R_{V-REx}(h) = \beta \cdot Var(\{R_1(h), ..., R_m(h)\}) + \sum_{e\in\mathcal{E}_{tr}} R_e(h) \tag{11}$$

The implementation for VREx on graphs should be straight forward since it is just a new regularized loss for a graph representation learner.

RICE: We describe here in full detail the implementation of RIA using the RICE regularizer and how RICE still fits the causal model we define in Section A.

Let the the support of a distribution be the subset of its domain where it has nonzero measure. This is denoted $supp(P) = \{x \in dom(P) | P(x) > 0\}$

Definition B.1. $P_{tr} := \sum_{e\in\mathcal{E}_{tr}: \sum_{e\in\mathcal{E}_{tr}} \lambda_e=1, \lambda_e\geq 0} \lambda_e \cdot P^e$ is the mixture of the training distributions with some λ_e from which it is possible to sample the training datasets $D_{tr} := \sqcup_{e\in\mathcal{E}_{tr}} D^e$, $D^e \subset supp(P^e)$ for $e \in \mathcal{E}_{tr}$. P_{tr} is conditional on D_{tr}.

RICE assumes a causal model. The causal model we define in Section A is compatible with the causal model of RICE. The causal model of RICE assumes that, given the data, the label is generated by the map $Y = m(c_*(X, A), \eta)$ where η is an exogenous variable, c_* coincides with the map we defined in Section A and m is any label producing map. RICE is formulated as a constrained optimization problem:

$$\min_\theta \mathbb{E}_{(G,Y)\sim P_{tr}}[l(h_\theta(G), Y)] \tag{12a}$$

$$\text{s.t. } h_\theta \circ T = h_\theta \forall T \in \mathcal{I}_{c_*}(supp(P_{tr}) \tag{12b}$$

where $\mathcal{I}_{c_*}(supp(P_{tr}))$ is defined below:

Definition B.2. (Causal Essential Invariant Transformations) (Wang et al., 2022)

$$\mathcal{I}_{c_*}(S) = \{T_i | c_*(X_1, A_1) = c_*(X_2, A_2) \Rightarrow \\ \exists T_1 ... T_k \text{ with } c_* \circ T_i = c_* \forall i, \text{ s.t.} \\ T_1 \circ ... \circ T_k(X_1, A_1) = (X_2, A_2) \\ \text{and } \forall (X_1, A_1), (X_2, A_2) \in S\} \tag{13}$$

We notice that a subset of the causal essential invariant transformations are just the invertible data augmentations which satisfy $c_* \circ T = c_*$. Implementing these data augmentations, such as edge addition and deletion on graphs, to approximate $\mathcal{I}_{c_*}(S)$ is simple and effective for graphs. We can thus narrow down the number of hyper parameters.

Proposition B.3. *The $\mathcal{I}_{c_*}(S)$ of Definition B.2 contains the set $\mathcal{I}^{inv}_{c_*}(S)$ of invertible transformations on data support S that satisfy $c_* \circ T = c_*$.*

Proof. We show that if T is invertible and satisfies $c_* \circ T = c_*$, then $T \in \mathcal{I}_{c_*}$(S).

We first show that the identities $\{I_{n_0}\}_{n_0\leq N}$, which depend on the number of graph nodes n_0, is in $\mathcal{I}_{c_*}(S)$. Let $(X_1, A_1) = (X_2, A_2)$ represent a graph of n_0 nodes, then we have that $c_*(X_1, A_1) = c_*(X_2, A_2)$ and that $I_{n_0}(X_1, A_1) = (X_2, A_2)$ for I_{n_0} the identity on (X_1, A_1).

For any $(X_1, A_1), (X_2, A_2) \in S$, $c_*(X_1, A_1) = c_*(X_2, A_2)$ then there exists $T' \in \mathcal{I}_{c_*}(P)$ s.t. $I_{n_0} \circ T'(X_1, A_1) = T^{-1} \circ T \circ T'(X_1, A_1) = (X_2, A_2)$. This shows that both T and T^{-1} are in $\mathcal{I}_g(S)$ for all T invertible over all graph sizes in the data support S.

□

Proposition B.3, tells us that we may use the invertible transformations on graphs such as edge deletion/addition in the regularization term of RICE. This means we can implement a regularizer for an OoD loss by the following OoD regularization term:

$$\textbf{OoD-Reg}_{RICE}(h_\theta, \{(G_w^e, Y^e)\}_{e\in\mathcal{E}_{tr}}) = \frac{\alpha}{n}\sum_{e=1}^{n}\mathbb{E}[\max_{T\in\mathcal{I}_{edge}^{inv}(\mathcal{G}_{X,A})} |(h_\theta \circ T(\mathbf{G}_w^e) - h_\theta(\mathbf{G}_w^e)|_2] \tag{14}$$

where Y^e is a set of labels for environment e, G_w^e is a set of adversarially augmented graphs for environment e and h_θ is a graph representation learner.

C. Hyperparameters and Dataset Information

Hyperparameters							
acc	CMNIST	SST2	Motif		AMotif		Synth
covariate	color	length	basis	size	basis	size	basis
lr	1e-3	1e-3	1e-3	1e-3	1e-3	1e-3	1e-3
lr_{adv}	1e-4	1e-4	1e-4	1e-4	1e-4	1e-4	1e-4
epochs	500	200	200	200	200	200	100
num. edge augs.	10	10	10	10	10	10	10
k	1	1	0	0	5	5	20
arch	GIN	GIN	GIN	GIN	GIN	GIN	GIN
num layers	5	5	3	3	3	3	2
p_{edge}^{add}	0.1	0.1	0.01	0.01	0.01	0.01	0.01
p_{edge}^{del}	0.1	0.1	0.01	0.01	0.01	0.01	0.01

Table 2. Superset of all hyper parameters shared across all datasets and shifts for all experiments.

We describe here some more information about each dataset we use in our experiments:

- CMNIST (Arjovsky et al., 2019) Dataset is derived from the MNIST dataset from computer vision. It is curated by (Gui et al., 2022). Digits are colored according to their domains. Specifically, in covariate shift split, we color digits with 7 different colors, and digits with the first 5 colors, the 6th color, and the 7th color are categorized into training, validation, and test sets.

- SST2 (Socher et al., 2013) Derived from a natural language sentiment classification dataset. Each sentence is transformed into a grammar tree graph, where each node represents a word with corresponding word embeddings as node features. The dataset forms a binary classification task to predict the sentiment polarity of a sentence. We select sentence lengths as domains since the length of a sentence should not affect the sentimental polarity.

- Motif (Wu et al., 2022b) Each graph in the dataset is generated by connecting a base graph and a motif, and the label is determined by the motif solely. Instead of combining the base-label spurious correlations and size covariate shift together as in (Wu et al., 2022b), the size and basis shifts are separated. Specifically, we generate graphs using five label irrelevant base graphs (wheel, tree, ladder, star, and path) and three label determining motifs (house, cycle, and crane). To create covariate splits, we select the base graph type and the size as domain features. There are no node attributes in this dataset.

- AMotif (a modification of Motif to have attributes) Taking the same graph structures from Motif, we use node attributes of dimension 256 all sampled from a $N(0,(e+1)^2)$, where e is the environment index. Covariate shifts are achieved by changing the basis or size as in Motif each shift indexed by some e.

- Synth We construct a synthetic dataset as described in Section 5. The dataset is a modification of Motif, which generates data by a joining operation between causal and spurious graphs. In our construction, we construct $(X_C, A), (X_S, A)$ as in AMotif. We let the joining operation be the map $(J_X(X_C, X_S), J_A(X_C, X_S)) = c_\xi^{-1}(X_C + X_S, A) = (X, A)$ where ξ are neural weights. We assume that the map c_ξ is invertible and has an inverse c_ξ^{-1} defined by a GIN neural network that maps from the graph $(X_C + X_S, A)$ to the graph $G = (X, A)$. GIN is not guaranteed to be injective, however it is a good enough approximation to one in practice. The label is defined by $Y = m(X_C, A) + \eta$ where m is a MLP and $\eta \sim N(0, \sigma(MLP(\tilde{e})))$ where $\tilde{e}$ is a one-hot encoding of the environment index and $\sigma \circ MLP$ is a fixed neural mapping to a tensor of numbers in $(0, 1)$. We can further assume that c_*, the causal map, can be obtained by $c_*(X, A) = c_\xi(X, A) - s_\xi(X, A)$ where c_* is deterministic and ξ is initialized by $\xi \sim N(0, MLP(\tilde{e})$. For the RIA-RICE implementation c_* is assumed to exist and allows us to obtain a solution of the form $\phi \circ c_*$. For RIA-IRM and RIA-VREx, so long as our data generation process coincides with the model of (Arjovsky et al., 2019) is satisfied, The distribution shifts are induced by varying $\tilde{e}$ and

thus affeting η and α simultaneously. There are 4 environments in $\mathcal{E}$. Two environments are combined together for training, the third for validation, and the remaining environments are for testing.

We list in Appendix-Table 2 the hyperparameters of our approachs on all datasets experimented with.