This CVPR 2021 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Adversarial Invariant Learning

Nanyang Ye
Shanghai Jiao Tong University
Shanghai, China
ynylincoln@sjtu.edu.cn

Jingxuan Tang
Peking University
Beijing, China
1800010657@pku.edu.cn

Huayu Deng
Shanghai Jiao Tong University
Shanghai, China
deng_hy99@sjtu.edu.cn

Xiao-Yun Zhou
PAII Inc
Rockville, USA
xiaoyun.zhou27@gmail.com

Qianxiao Li
National University of Singapore
Singapore
qianxiao@nus.edu.sg

Zhenguo Li
Huawei Noah's Ark Lab
Hong Kong SAR
li.zhenguo@huawei.com

Guang-Zhong Yang
Shanghai Jiao Tong University
Shanghai, China
Guang-Zhong.Yang@aaas.org

Zhanxing Zhu(✉)
Peking University
Beijing, China
zhanxing.zhu@pku.edu.cn

Abstract

Though machine learning algorithms are able to achieve pattern recognition from the correlation between data and labels, the presence of spurious features in the data decreases the robustness of these learned relationships with respect to varied testing environments. This is known as out-of-distribution (OoD) generalization problem. Recently, invariant risk minimization (IRM) attempts to tackle this issue by penalizing predictions based on the unstable spurious features in the data collected from different environments. However, similar to domain adaptation or domain generalization, a prevalent non-trivial limitation in these works is that the environment information is assigned by human specialists, i.e. *a priori, or determined heuristically. However, an inappropriate group partitioning can dramatically deteriorate the OoD generalization and this process is expensive and time-consuming. To deal with this issue, we propose a novel theoretically principled min-max framework to iteratively construct a worst-case splitting, i.e. creating the most challenging environment splittings for the backbone learning paradigm (e.g. IRM) to learn the robust feature representation. We also design a differentiable training strategy to facilitate the feasible gradient-based computation. Numerical experiments show that our algorithmic framework has achieved superior and stable performance in various datasets, such as Colored MNIST and Punctuated stanford sentiment treebank (SST). Furthermore, we also find our algorithm to be robust even to a strong data poisoning attack. To the best of our knowledge, this is one of the first to adopt differentiable environment splitting method to enable stable predictions across environments without environment index information, which achieves the state-of-the-art performance on datasets with strong spurious correlation, such as Colored MNIST.*

1. Introduction

Most machine learning algorithms rely on the assumption that the training data and test data are sampled from the same distribution. This poses a fundamental problem as in real scenario where machine learning systems have to make decisions based on data sampled from unseen distributions, *i.e.* out-of-distribution (OoD) data. In conventional machine learning tasks, we collect a large dataset and randomly split it into training and test parts, thus naturally forcing the two to follow the same distribution. This procedure, while convenient, can lead to an overly optimistic estimation of the true generalization error. Indeed the model can over-fit to this original distribution and poorly generalize to a new test set collected in a slightly different environment. The root of this limitation is the tendency for models to use spurious environmental variations as discriminative features, since minimizing the empirical risk forces the models to exploit all features including spurious features from data that are not related to the explanation of the results. For example, as shown in Figure 1, the spurious correlation between the class sheep and the grassland background may lead to worsen generalization performance on test sets

where the background are desserts or highways, which also reflects the Occam's razor that the model tends to learn the most simple rule for prediction.

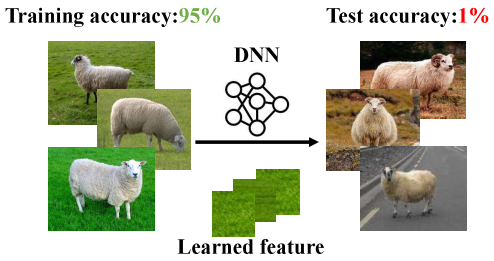

Figure 1. Example of spurious correlation.

To address this issue, several methods have been proposed. Domain generalization approaches have demonstrated good performance when testing on data with different styles, such as photo or painting [5, 19, 20, 9]. However, as shown in our experiments and [3], domain generalization or adaptation methods can still easily fall into the correlation-versus-causation trap because the common design principle is to force the predictor to learn the contours instead of the textures which is highly problem dependent. As shown in our experiments, these methods are unable to eliminate the effects of spurious features. On the other hand, methods based on classic causal inference methods, such as deep global balancing regression strategy [17], and causal inference via invariant prediction [24] have been proposed. However, they are only tested on simple low dimensional regression tasks where the correlation between each feature is not conspicuous. It is still unknown whether they can generalize well on typical computer vision tasks where image pixels are highly correlated.

Recently, to tackle the correlation-versus-causation dilemma, Shiori *et al*. proposed a distributionally robust approach to optimize the loss on the worst-case environment where data with different properties are assumed to be split into different environments [26]. Later, Martin *et al*. demonstrated that optimizing on the worst-case environment could still lead to solutions over-fitted to spurious features. Instead, the invariant risk minimization (IRM) was introduced by monitoring the invariance across different environments [3]. In IRM, a regularization term is introduced to avoid the performance degeneration in causal inference methods caused by eliminating features, such as global balancing in [17]. Although algorithms like IRM look promising, similar to typical methods in domain adaptation or domain generalization, data need to be manually split into different environments to enable them to minimize the performance discrepancies across environments. This fundamental restriction prevents these schemes to be practical. First, how data are split can largely influence the final generalization abilities. As shown in Figure 3, environment misspecification can lead to degenerated performances in IRM. Second, sometimes, it is impossible to split data in continuous environments. For example, it is hard to decide whether grouping photos collected at dawn together with photos collected in the day time or night time.

In this paper, based on theoretical analysis, we propose a novel approach, adversarial invariant learning (AIL), to iteratively construct worst-case environments/group splittings for IRM to learn the invariant features for prediction. This optimized partitioning over the training samples forces the network to train towards the robust representation across this worst-case grouping, thus benefiting the OoD generalization. We demonstrate that the combination of the adversarial partitioning and IRM is the key for eliminating effects of spurious features and also being robust to data poisoning attacks, where data poisoning attack aims at breaking the learning process by adding maliciously perturbed data which adds wrong correlations between the data and labels. The contributions can be summarized as follows:

1. Based on theoretical analysis, we formulate the OoD generalization problem as maximizing the lowerbound of the log likelihood of causal predictions.

2. For practical instantiation of AIL, we propose to adopt unsupervised learning methods, such as variational autoencoders (VAE, [15]) and K-means clustering to automatically partition the data into different groups based on the latent features learned in the initial phase of the algorithm. This is followed by applying IRM to learn an invariant predictor based on these newly generated data splits.

3. We unify the unsupervised and supervised learning component with a min-max formulation and propose a differentiable approach for updating VAE during training to dynamically create challenging environments for the supervised learning component. Ablation studies show the adversarially trained VAE can significantly improve the OoD generalization performance.

4. We demonstrate that our algorithm can achieve superior empirical performance compared with existing alternatives in various benchmark datasets, such as Colored MNIST and Punctuated Stanford Sentiment Treebank (SST). We also show that our approach is robust to data poisoning attacks.

2. Preliminary

We present a formal description of the problem setting. First, as shown in Figure 2, we consider a data generating process in the form of structured causal model, where $\mathbf{X}$ is the collected data (e.g. images), $\mathbf{T}$ is the environment variable, and $\mathbf{S} = f(\mathbf{X}, \mathbf{T})$ represents the semantic features

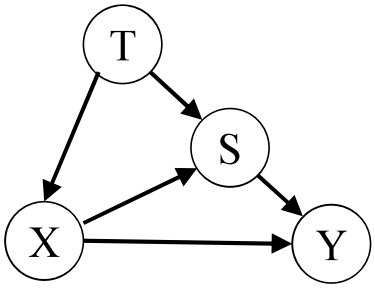

Figure 2. Data generating process. The semantic feature S is both affected by observable variable $\mathbf{X}$ and environment variable $\mathbf{T}$. The correlation between label $\mathbf{Y}$ and observable variable $\mathbf{X}$ can be affected by $\mathbf{T}$, thus leading to over-fitting.

resulted from a non-linear combination of $\mathbf{X}$ and $\mathbf{T}$ determining the label $\mathbf{Y}$. From this model, $\mathbf{T}$ both affects $\mathbf{X}$ and $\mathbf{Y}$ which is the cause for over-fitting to spurious environment features. For example, for image classification tasks, most images of cars may have highway backgrounds. The background information is the spurious feature brought by the data collection environment. It has already been demonstrated that the generalization abilities of deep neural networks (DNN) trained on ImageNet can be severely impaired by changes in the data collection process [25], e.g. the background. Given a collected dataset $\mathcal{S} = (\mathbf{x}_i, y_i)_{i=1}^N$, the objective is to train a DNN $h_{\boldsymbol{\theta}} : \mathcal{R}^d \to \mathcal{R}^\Omega$ for Ω-class classification that generalizes well under different environments.

Problem of empirical risk minimization. To formally show the over-fitting issue of commonly-used empirical risk minimization, we consider the conditional distribution of $p(\mathbf{x}|t)$, $t \sim p(t)$ is the prior distribution of the environment. Classical empirical risk minimization (ERM) tries to minimize the following risk function:

$$\begin{aligned}\mathcal{R}_{\text{ERM}}(\boldsymbol{\theta}) &= \mathbb{E}_{\mathbf{X},\mathbf{Y}}[\ell(h_{\boldsymbol{\theta}}(\mathbf{X}), \mathbf{Y})] \\ &= \int \ell(h_{\boldsymbol{\theta}}(\mathbf{x}), y) p(\mathbf{x}, y) \mathrm{d}\mathbf{x}\mathrm{d}y \\ &= \int \ell(h_{\boldsymbol{\theta}}(\mathbf{x}), y) p(y|\mathbf{x}) p(\mathbf{x}|t, \mathbf{y}) p(t) p(\mathbf{y}) \mathrm{d}\mathbf{x}\mathrm{d}t\mathrm{d}y \end{aligned} \tag{1}$$

From this equation, the distribution of $p(t)$ can heavily influence the loss landscape with regard to $\boldsymbol{\theta}$. However, $p(t)$ may be quite different in training and test dataset. It is demonstrated in [3] and confirmed in our experiment section that manipulating $p(t)$ can generate OoD distributions and even mislead the model to rely completely on environmental spurious features for prediction.

Challenging settings. In previous works, the t labels containing environment information are assumed to be known. The above problem may be mitigated by monitoring the changes in ERM loss across environments. However, in our setting, t is a hidden variable that cannot be directly observed. The spurious correlation between t and Y can be misleading for most methods relying on correlation. A naive solution might be assigning a random t to each data point or label them heuristically. However, this misspecification can lead to suboptimal performance as shown in Figure 3.

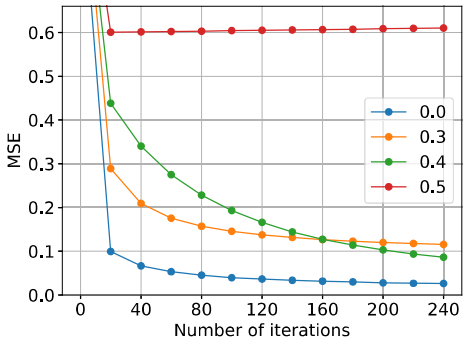

Figure 3. Performance of IRM under different rates of environment misspecification. Y-axis indicates the MSE between the current solution and the true invariant solution across different environments. Higher rate of misspecification can lead to failure of convergence. When the rate is 0.5, two environments' data are uniformly mixed. More details are in the Appendix.

3. Proposed method

3.1. Theoretical analysis

From the above example, using the correlation-based prediction $P(y|\mathbf{x}; \boldsymbol{\theta})$ can elicit severe over-fitting to spurious features. To derive the causal prediction $P(y|do(\mathbf{x}); \boldsymbol{\theta})$ that excludes the effects of environment variable T, we first use the backdoor adjustment method [23] to translate the causal inference problem in the context of classic statistical learning:

Theorem 1. ***(Backdoor adjustment).*** *Given the structured causal graph in Figure 2 and neural network parameters* $\boldsymbol{\theta}$, $p(y|do(\mathbf{x}); \boldsymbol{\theta}) = \sum_t p(y|\mathbf{x}, S = f(x, t); \boldsymbol{\theta}) p(t)$.

The proof of this theorem can be found in the Appendix. From this theorem, if $p(t)$ is known, we may be able to calculate the causal prediction given image x. However, as discussed before, only $p_{train}(t)$ is known for training set and $p_{test}(t)$ can be quite different. To achieve robust generalization performance across possible variations of $p(t)$, one may consider optimizing only the worst distribution to maximize $p(y|do(x))$ drawn from $p(t)$ whereas this might lead to over-fitting as only a small fraction of data can be used for training. Instead, we consider maximizing the lower bound

of $p(y|do(x);\boldsymbol{\theta})$ by decomposing $p(t)$ into series of distributions that can be summed to the original distribution. We name each of the distribution *sub-distribution* for clarity. We first make the following assumption:

Assumption 1. *The distribution of t in the training data can be decomposed into K sub-distributions, which satisfies $p_{train}(t) = \frac{1}{K}\sum_{k=1}^{K} p^k_{train}(t)$.*

This assumption assumes $p_{train}(t)$ is separable. We denote the series of sub-distributions $\{p^k_{train}(t)\}$ as $\mathcal{P}$. Then, we show the lower bound of likelihood can be represented by the weighted sum of likelihood of each sub-distribution:

Theorem 2. ***(Lower bound of likelihood).*** *If Assumption 1 holds and the Rényi divergence between $p_{train}(t)$ and $p(t)$ satisfies: $D_\infty\left(p(t)||p_{train}(t)\right) \leq C$, then:*

$$\log p(y|do(x);\boldsymbol{\theta})] = \log \sum_t p(y|x, S = f(x,t);\boldsymbol{\theta})p(t)$$
$$\geq \min_{\mathcal{P}} \left(\sum_{k=1}^{K} \frac{\sum_t \log p(y|x, S = f(x,t);\boldsymbol{\theta}) p^k_{train}(t)}{K \exp^C} \right) \tag{2}$$

The proof of this theorem can be found in the Appendix. This theorem indicates that maximizing the likelihood of $p(y|do(x);\boldsymbol{\theta})$ can be alternatively achieved by maximizing its lower bound via finding series of sub environment variable distributions that elicits the worst possible performance on the prediction. Note that the constant C determines how close is the lower bound to the true log likelihood. This theorem provides a way to obtain the causally robust predictor. We name this approach ***"Adversarial Invariant Learning"*** (AIL). Then the loss function of AIL $\ell(\boldsymbol{\theta})$ can be written as a min-max formulation:

$$\min_{\boldsymbol{\theta}} \max_{\mathcal{P}} \sum_k \frac{1}{K} \mathbb{E}[\sum_t \log p(y|x, S = f(x,t);\boldsymbol{\theta}) p^k_{train}(t)]$$
$$+ \lambda R(\boldsymbol{\theta}, \mathcal{P}) \tag{3}$$

where λ is the coefficient for regularization function $R(\boldsymbol{\theta}, \mathcal{P})$ also absorbing the constant $1/\exp^C$. Next, we will provide a reasonable instantiation of AIL as below.

3.2. Instantiation of AIL

The proposed method consists of an unsupervised component for creating challenging sub-distributions of data and a supervised learning component for learning an invariant predictor that evolves together via an adversarial training scheme. In the unsupervised module, we model the inner data structure with a generative model (i.e. variational autoencoder) pre-trained on the unlabeled dataset, and then later fine-tune with adversarial gradients to maximize loss function to dynamically create challenging "assignments" (e.g. by K-means clustering) of environments for the supervised learning component. We argue that the combination of unsupervised and supervised module is necessary. As the spurious correlation between irrelevant features and labels are often the cause for degenerated generalization performance on OoD data. In the supervised part, IRM is used as the regularization function to achieve an invariant predictor across the environment splitting by the adversarially trained VAE and clustering. It is worth noting that this is not the only way to implement AIL. For example, the K-means clustering can be substituted by Gaussian mixture model. The invariant minimization regularization can also be replaced by other methods, such as risk extrapolation (ReX, [16]). We leave more detailed discussions of the module choice in the Appendix.

3.2.1 Bayesian variational clustering

In this module, we use the semantic representation learned from a variational autoencoder (VAE, [15]) and conduct K-means clustering within each class to split data into different sub-groups.

Variational autoencoder We use VAE to model the data generating process $p(\mathbf{x}|t)$. In VAE, given an input datum $\mathbf{x}$, it is first encoded into the latent code $\mathbf{g}$, and then the latent code $\mathbf{g}$ is decoded by DNN to reconstruct the input datum. To train VAE, the negative evidence lower bound (ELBO) is used as the loss function:

$$\ell_{\text{ELBO}}(\boldsymbol{\theta}_{\text{VAE}}) = -D_{\text{KL}}(q(\mathbf{g}) \parallel p_0(\mathbf{g})) + \mathbb{E}_{q(\mathbf{g})}[\log p(\mathbf{x}|\mathbf{g})] \tag{4}$$

where D_{KL} is the Kullback–Leibler divergence, $p_0(\mathbf{g})$ is the prior distribution of the latent code set to be a unit Gaussian, and the second term is the reconstruction loss, $q(\mathbf{g})$ is the variational distribution for the true posterior of $p(\mathbf{g}|\mathbf{x})$, which follows a Gaussian distribution with parameters determined by VAE. The idea of VAE is to use an approximate distribution $q(\mathbf{g})$ to model the latent distribution of the data generating process $p(t|\mathbf{x})$. The learned latent semantic distribution $q(\mathbf{g})$ for input data $\mathbf{x}$ is then used for inner-class differentiable K-means clustering.

Inner-class differentiable K-means clustering Based on the learned latent codes $\mathbf{g}$, we use K-means clustering within each class to split data into different environments. We assumed the data can be split into T clusters for simplicity. Then, we conduct T clusters K-means clustering **within each class** for C classes. We found the inner-class clustering is necessary as data with different classes pose distinct distributions that may cover the effects of environment variable changes.

Implementing a differentiable clustering method is non-trival. To allow adversarial gradient calculation for VAE

(described in Section 3.2.3), we modify the original K-means clustering to a differentiable version. Instead of assigning an integer to denote the class index, we calculate the probability of a given datum $\mathbf{x}$ based on the cluster centers,

$$\alpha(t|\mathbf{x}) = \frac{\exp\left(d_t(\mathbf{x}) - d_{max}(\mathbf{x})\right)}{\sum_{t=1}^{T}\exp\left(d_t(\mathbf{x}) - d_{max}(\mathbf{x})\right) + \epsilon}, \quad \text{where } t = 1, \cdots, T \tag{5}$$

where $\alpha(t|\mathbf{x})$ is the probability of a datum $\mathbf{x}$ belonging to environment t, $d_t(\mathbf{x})$ is the distance of the latent code of the datum to the t-th cluster center, $d_{max}(\mathbf{x})$ is the maximum distance, and ϵ is the added parameter to improve numerical stability. Through adversarial VAE updates, the clustering step aims at creating worst possible data splittings for later algorithmic paradigm to learn. Another advantage of the adversarial updates is that the T can be adaptively determined as when T is too large, some clusters will have zero members in order to maximize the loss. Note that for saving computational time, we use this differentiable K-means instead of Gaussian mixture model (GMM) as GMM needs a few iterations to train before clustering every time. We thus leave it for future work for more complex clustering methods, such as GMM.

3.2.2 Invariant Risk Minimization Regularization

Based on the inferred $\alpha(t|\mathbf{x})$, we are able to apply the IRM to learn an invariant predictor across different environments. Recently, Martin *et al.* [3] proposed the IRM method to achieve invariant prediction across environments which is arguably the hallmark for causal inference [7, 1]. In IRM, given data collected in T environments, it aims to learn a stable feature extractor $\Gamma : \mathbf{X} \to \mathbf{H}$ to transform data $\mathbf{X}$ into a feature space $\mathbf{H}$ that is invariant across different environments and a classifier $\bar{\mathbf{w}} : \mathbf{H} \to \mathbf{Y}$ to predict based on the feature. Formally, it can be written as:

$$\min_{\substack{\Gamma:\mathbf{X}\to\mathbf{H}\\ \bar{\mathbf{w}}:\mathbf{H}\to\mathbf{Y}}} \sum_{t=1}^{T} R^t(\bar{\mathbf{w}}\circ\Gamma) \quad \text{s.t. } \bar{\mathbf{w}} \in \operatorname{argmin}_{\bar{\mathbf{w}}^t} R^t(\bar{\mathbf{w}}^t\circ\Gamma), \text{ for } t = 1, \cdots, T \tag{6}$$

where R^t is the ERM risk of the t-th environment. However, there are several problems that may prevent IRM from being practically useful:

1. Feasibility of solutions. This problem may not be feasible and be hard to solve as requiring $\bar{\mathbf{w}}$ to be simultaneously optimal for all environments is too demanding.

2. Multiple solutions. There might be multiple solutions as well, as $\bar{\mathbf{w}}\circ\Psi\circ\Psi^{-1}\circ\Gamma$ will also be a solution when Ψ is an invertible transform. Instead, IRM sets $\bar{\mathbf{w}}\circ\Gamma$ as $(w\cdot\tilde{\mathbf{w}})\circ\Gamma$ by multiplying a scalar w to the classifier $\tilde{\mathbf{w}}$. As $\tilde{\mathbf{w}}$ can be absorbed in Γ, satisfying the condition in Eq. 6 simply requires solving multiple optimization problems.

Instead, in [3], the authors proposed to add a regularization term $\|\nabla_{w|w=1}R^t(w\cdot\Phi)\|^2$ to the original ERM loss, where $w \in \mathcal{R}$ and $\Phi : \mathbf{X} \to \mathbf{Y}$ is the parameters of the neural network. [1]:

$$\min_{\Phi:\mathbf{X}\to\mathbf{Y}} \sum_{t=1}^{T}\left(R^t(\Phi) + \lambda\|\nabla_{w|w=1}R^t(w\cdot\Phi)\|^2\right) \tag{7}$$

Note that this changes the original bi-level optimization problem into a single level one with regard to model parameters Φ. Now, based on the inner-class differentiable K-means clustering result, we subsitute the original ERM loss with the first term in Eq. 3:

$$\min_{\Phi:\mathbf{X}\to\mathbf{Y}} \sum_{t=1}^{T}\mathbb{E}_{\mathbf{X}^t,\mathbf{Y}^t,\boldsymbol{\alpha}^t}[\ell(h_\Phi(\mathbf{X}),\mathbf{Y})] + \lambda\|\nabla_{w|w=1}\mathbb{E}_{\mathbf{X}^t,\mathbf{Y}^t,\boldsymbol{\alpha}^t}[\ell(w\cdot h_\Phi(\mathbf{X}),\mathbf{Y})]\|^2 \tag{8}$$

where $\alpha(t|\mathbf{X})$ is the probability of the data belonging to the t-th environment calculated in Eq. (5), $\mathbf{X}^t$ and $\mathbf{Y}^t$ are data and labels split to the t-th environment.

3.2.3 Formulation of Adversarial Invariant Learning

Motivated from Theorem 2, to learn an invariant predictor that can exclude spurious correlation in the datasets, we consider a min-max game between an attacker and a learner to maximize the lowerbound of the log likelihood of the causal prediction. In this game, the attacker tries to maximize the loss by creating hard data splittings to mimic OoD distributions and while the learner attempts to minimize the loss to learn predictor that are robust to distribution shifts. More specifically, the attacker is the adversarially trained VAE and clustering in the unsupervised learning component and the learner is the supervised DNN. As the distribution of environment splitting $\alpha(t|\mathbf{x})$ is determined by VAE, we denote $\alpha(t|\mathbf{x})$ as $\alpha_{\text{VAE}}(t|\mathbf{x})$, the objective function of this min-max game is then,

$$\min_{\Phi:\mathbf{X}\to\mathbf{Y}}\max_{\theta_{\text{VAE}}}\mathbb{E}_{\mathbf{X}^t,\mathbf{Y}^t,\boldsymbol{\alpha}^t_{\text{VAE}}}[\ell(h_\Phi(\mathbf{X}),\mathbf{Y})] + \left(\sum_{t=1}^{T}\lambda\|\nabla_{w|w=1}\mathbb{E}_{\mathbf{X}^t,\mathbf{Y}^t,\boldsymbol{\alpha}^t_{\text{VAE}}}[\ell(w\cdot h_\Phi(\mathbf{X}),\mathbf{Y})]\|^2\right) \tag{9}$$

Through this min-max formulation, we cast the original problem of learning non-spurious structure from a single dataset into a robust optimization problem to find a model

[1] The proof why fixing w as 1 is enough is in [3]

parameter Φ that can achieve good performance on worst possible data splittings. This formulation could enforce the classifier to be robust to the inaccurate splitting of the environments. We will show in the ablation studies that this formulation can significantly improve the generalization abilities under a large distribution gap between training and test environments in Section 4.4.

3.2.4 Algorithm

To solve the game formulated in Eq. (9), we propose the procedure described in Alg. 1 in the Appendix. We first pretrain VAE in the first several epochs and then use stochastic gradient optimization with min-max optimization to achieve equilibrium in the game.

Comparison with existing works Compared with distributional robust optimization (DRO, [28]), our approach optimizes for worst possible splitting of data into different environments to improve the robustness, while DRO methods only consider the worst-case perturbation near the original data distribution. Deep global balancing regression (DGBR, [17]) uses the bottleneck representations of an autoencoder (AE) as the input features for a multilayer perceptron to predict labels and optimize at the same time for reconstruction and prediction loss. This constrains the generalization ability as the AE's features are learned for reconstructing the inputs which may be quite different from features useful for discriminative tasks. The domain generalization by solving jigsaw puzzles (JIGSAW, [6]) introduces an auxiliary task by randomly scrambling image patches and then predicting the correct ordering sequence to reconstruct the original image (JIGSAW puzzle). However, solving the JIGSAW puzzle can only help the model to generalize well when the distribution shift is not large [14]. And thus, different from these approaches, our method in principle achieves the generalization across multiple test environments without knowing the environment index *a priori*. More importantly, through adversarial learning over the environment splitting, our approach exhibits strong robustness to malicious clustering, as shown in the experiment part.

4. Experiments

We compare AIL (proposed), invriant risk minimization (IRM [3]), distributional robust optimization (DRO [28]), deep global balancing regression (DGBR [17]), domain generalization by solving jigsaw puzzles (JIGSAW [6]), and empirical risk minimization (ERM) methods. DGBR is a widely used baseline method for causal inference in the deep learning setting. JIGSAW is the state-of-the art work for domain generalization research and it is one of the very few methods that do not require domain (environment) index. We evaluate these approaches on colored MNIST dataset, punctuated SST-2 dataset, and data poisoning attacked MNIST dataset[2]. We also leave more baseline comparison experiments including recently published papers and detailed experiment settings in the Appendix.

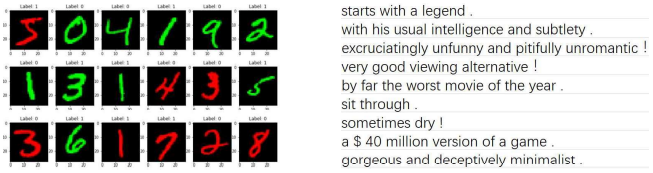

Figure 4. **Left:** Colored MNIST examples. Color is the spurious feature. **Right:** Punctuated SST-2 examples. Mark is the spurious feature. Better viewed when zoomed in.

4.1. Image recognition tasks

In this section, we conduct experiments based on the new constructed MNIST dataset with a similar experiment protocol as in IRM[3]. The new dataset consists of data generating from three environments (two randomly mixed for training, one for test). Similar to the setting in [3], we follow these steps to generate the colored MNIST datasets for each environment: first, we assign a binary label $\hat{y}$ to the image ($\hat{y} = 0$ for digits 0-4 and $\hat{y} = 1$ for 5-9). We then flip labels with probability p. The color type indexes are determined by flipping $\hat{y}$ with probability e. These steps are the same for generating both the training and test environments but with different e values, which denote the environment shift between training and test. For the training environments, we set $e = 0.1$ and $e = 0.2$. For the test environment, we set $e = 0.9$. The difference of e in the training and test data poses great challenges for commonly-used machine learning algorithms as i.i.d assumptions cannot be satisfied. In this case, $\mathbf{z}$ is the strokes of digits, which is the key feature determining the true labels of digits. t is the spurious feature color, which disturbs the model. We can adjust p to control the correlation strength of the spurious feature color to the label.

We train DNNs on the mixed colored MNIST dataset with different methods and visualize the results in Figure 5. We can observe that with the increase of p, AIL has decreased training accuracies but maintains a relatively stable high accuracy on the test dataset that follows different distribution as the training dataset. This indicates that AIL is able to react to the distribution shift in the training dataset and generalize well despite the effects of spurious features. Note that DGBR achieves quite stable performances as it utilizes features generated by an autoencoder. Although

[2]Note that there are some quite recent work we wish to compare [12]. However, until the submission of this paper, the released code has not been ready yet.

[3]https://github.com/facebookresearch/InvariantRiskMinimization

DGBR improves the stability of predictions as in [17], it heavily restricts the DNN's generalization abilities. On the other hand, domain generalization methods such as JIGSAW cannot deal with this situation, as they require the distance between the distribution of the target domain and the training domain to be small [14].

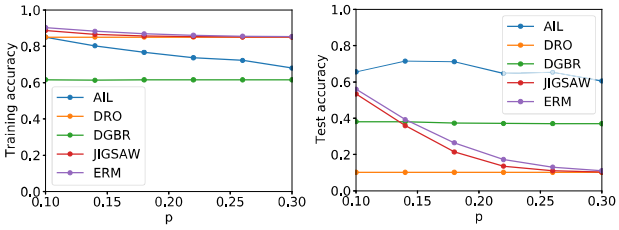

Figure 5. **Left:** training accuracy. **Right:** test accuracy. When p gets larger, the distribution difference between the training and test dataset becomes larger. Better viewed when zoomed in.

4.2. Natural language processing tasks

Overly exploiting spurious features also results in a large performance gap in natural language processing tasks [11, 21, 22]. Here we conduct an experiment using the same setting as in [8] on the SST-2 dataset. The task is to predict whether a given sentence expresses positive or negative feelings in the SST-2 dataset as shown in Figure 4 Right. The dataset consists of five environments (two for training and three for test). Similarly, for generating data in each environment, all punctuation marks in the middle of sentences are removed, and then the labels are flipped with a probability e. Next, sentences are paired with a punctuation mark: positive with a period (.) and negative with an exclamation mark (!). Finally, with probability p, labels are flipped. Our implementation is based on [4]. In this experiment, as sentences can have variable lengths, a pre-trained vocabulary is used to embed each word in a sentence into a vector living in a compact high-dimensional manifold. In this task, we only evaluate ERM because the sum of the embedded feature is not meaningful for the autoencoder in DGBR. Likewise, DRO and JIGSAW are typically designed for image processing tasks.

The results are shown in Table 1. From Table 1, we can observe that ERM overly exploits all features including the spurious feature mark to achieve good performance on the training sets but with poor generalization performance on the test sets. On the other hand, AIL achieves relatively stable predictions across all environments.

4.3. Adversarial robustness to data poisoning attack

Data poisoning attack is an attack method in the training phase of machine learning models, where the attacker adds malicious examples ("poisoned" examples) into the training dataset to change the behaviour of the trained model at test time [13]. In this experiment, we focus on the case where the attacker's objective is to destroy the model's performance on the test dataset. Besides, to make the task even more challenging, similar to the setting in [27], we initialize the models with weights trained on the clean dataset and then fine-tune the models' last fully-connected layers on the poisoned dataset. We evaluate the robustness of different methods on the MNIST dataset. From the results shown in Table 2, we can observe that AIL exhibits robustness to even strong data poisoning attacks. The reason is as follows. Different from ERM that naively minimizes the total loss on the training set, AIL can split the polluted data into another environment to avoid over-fitting. It is worth noting that JIGSAW is also robust to this attack but still more vulnerable than AIL. This is because JIGSAW introduces an auxiliary task to ensemble the scrambled image patches, which prevents over-fitting the poisoned data.

[4] https://github.com/kakaobrain/irm-empirical-study

Table 1. Punctuated SST-2 dataset results. $1-e$ is the correlation between the spurious feature-mark and the label. The setting of e in the training and test environments are: $e_1 = 0.2$, $e_2 = 0.1$, $e_3 = 0.7$, $e_4 = 0.8$, $e_5 = 0.9$.

Settings	ERM	AIL
Training set (e_1)	**82.16%**	*60.45%*
Training set (e_2)	**90.57%**	*63.99%*
Test set (e_3)	32.77%	***49.32%***
Test set (e_4)	28.52%	***50.90%***
Test set (e_5)	15.72%	***43.48%***

Table 2. Adversarial robustness to data poisoning attack

Methods	Training accuracy	Test accuracy
ERM	72.01%	75.94%
DRO	52.41%	53.12%
DGBR	**89.72%**	50.58%
JIGSAW	86.00%	94.10%
AIL	*89.16%*	***97.05%***

4.4. Ablation study and discussion

Adversarial training For ablation studies, we first investigate the importance of adversarial splitting. We use the colored MNIST experiment as an example. The results are shown in Table 3. We denote the AIL without adversarial splitting as AIL(w/o adv). We can see that under various settings of p, adversarial training can stably improve the generalization performance. This is because adversarial training can dynamically explore hard environments to learn an invariant predictor across different environments.

To visualize the effects of adversarial training on data splitting, we plot the clustering results of pre-trained VAE and adversarially trained VAE on the same data in Figure 6. We can observe that adversarially trained VAE attempts to split the data in the worst possible way to train an invariant

Table 3. Ablation study: effect of adversarial training.

Test accuracies under different settings of p.			
Methods	$p = 0.1$	$p = 0.2$	$p = 0.3$
AIL(w/o adv)	65.04%	61.05%	55.47%
AIL	**65.58%**	**68.49%**	**60.53%**
Test accuracies under different settings of offsets β.			
Methods	$\beta = 0.001$	$\beta = 0.002$	$\beta = 0.003$
AIL(w/o adv)	60.77%	59.64%	58.48%
AIL	**62.41%**	**62.05%**	**66.91%**

predictor that is forced to be effective across environments and robust to environment assignments.

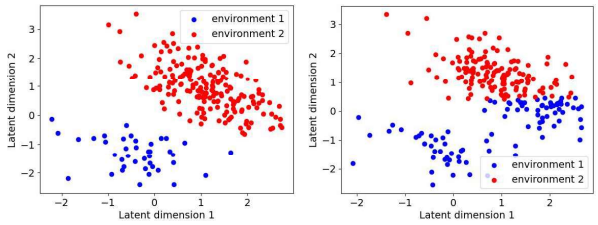

Figure 6. **Left:** clustering after pre-training VAE. **Right:** clustering after adversarially training VAE for one iteration.

Robustness to VAE parameters perturbations To check whether adversarially trained VAE can exhibit robustness to perturbations of VAE parameters, we add a constant offset β to all VAE parameters after pre-training and report the results in Table 3. As discussed in Section 3.2.1, there are multiple solutions for optimizing the ELBO loss in VAE. Different offsets at the initialization of VAE training can elicit different solutions that are not necessarily all beneficial for learning an invariant predictor. From Table 3, without adversarial training to adaptively update VAE, performance can be greatly degenerated. On the contrary, adding perturbations to the AIL may even lead to improved performances in some cases.

Robustness to cluster number misspecification Next, we discuss the selection of the number of clusters T in Bayesian variational clustering. The minimum number of clusters T is two. We increase T from two and observe that the test accuracy will be stable under the increasing T as shown in Table 4. This is because the min-max formulation in our implementation naturally incorporates an adaptive scheme to maintain a reasonable number of clusters to maximize the loss.

Comparison with IRM We also compare the performance of AIL and the original IRM with environment information prior assigned by humans on Colored MNIST. For fair comparisons, we adopt similiar test protocols as in [10]. The mean and standard deviation result is shown in Figure 7. From Figure 7, we can observe that AIL consistently outperforms IRM across different difficulty levels.

Table 4. Robustness to cluster number misspecification

Cluster number	Test accuracy	Cluster number	Test accuracy
T=2	66.76%	T=6	62.79%
T=3	69.23%	T=7	66.28%
T=4	66.79%	T=8	67.75%
T=5	66.34%	T=9	63.48%

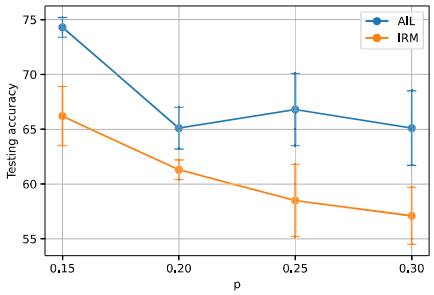

Figure 7. Comparison between AIL and IRM.

5. Conclusion and future work

In this paper, we have proposed AIL as one of the first algorithmic frameworks to adopt differentiable environment splitting method to enable stable predictions across environments without environment index information. AIL indicates a new promising research direction for OoD generalization. For future work, we will explore more generative models to effectively represent features in the high dimensional space and further improve the generalization performances.

Acknowledgements

Nanyang Ye was supported in part by National Key R&D Program of China 2017YFB1003000, in part by National Natural Science Foundation of China under Grant (No. 61672342, 61671478, 61532012, 61822206, 61832013, 61960206002, 62041205), in part by the Science and Technology Innovation Program of Shanghai (Grant 18XD1401800, 18510761200), in part by Shanghai Key Laboratory of Scalable Computing and Systems.

Zhanxing Zhu was supported by Beijing Nova Program (No. 202072) from Beijing Municipal Science & Technology Commission, and National Natural Science Foundation of China (No.61806009 and 61932001), PKU-Baidu Funding 2019BD005.

References

[1] Kartik Ahuja, Karthikeyan Shanmugam, Kush R. Varshney, and Amit Dhurandhar. Invariant risk minimization games,

2020. 5

[2] Anonymous. Systematic generalisation with group invariant predictions. In *Submitted to International Conference on Learning Representations*, 2021. under review. 12

[3] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2019. 2, 3, 5, 6, 12, 13

[4] Haoyue Bai, Rui Sun, Lanqing Hong, Fengwei Zhou, Nanyang Ye, Han-Jia Ye, S. H. Gary Chan, and Zhenguo Li. Decaug: Out-of-distribution generalization via decomposed feature representation and semantic augmentation, 2020. 13

[5] Fabio Maria Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2224–2233, 2019. 2

[6] Fabio Maria Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *CVPR*, 2019. 6, 13

[7] Nancy Cartwright. Two theorems on invariance and causality. *Philosophy of Science*, 70(1):203–224, 2003. 5

[8] Yo Joong Choe, Jiyeon Ham, and Kyubyong Park. An empirical study of invariant risk minimization, 2020. 7

[9] Qi Dou, Daniel C. Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *NeurIPS*, 2019. 2, 13

[10] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization, 2020. 8, 12

[11] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. 7

[12] Zeyi Huang, Haohan Wang, Eric P. Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *ECCV*, 2020. 6, 12

[13] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 19–35, 2018. 7

[14] Fredrik D. Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 527–536. PMLR, 16–18 Apr 2019. 6, 7

[15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013. 2, 4

[16] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex), 2020. 4, 12, 13

[17] Kun Kuang, Ruoxuan Xiong, Peng Cui, Susan Athey, and Bo Li. Stable prediction across unknown environments. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2018. 2, 6, 7

[18] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. *CoRR*, abs/1710.03077, 2017. 13

[19] Massimiliano Mancini, Zeynep Akata, Elisa Ricci, and Barbara Caputo. Towards recognizing unseen categories in unseen domains. In *ECCV*, August 2020. 2, 13

[20] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains, 2019. 2

[21] Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. 7

[22] Timothy Niven and Hung-Yu Kao. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664, Florence, Italy, July 2019. Association for Computational Linguistics. 7

[23] J. Pearl, Madelyn Glymour, and N. Jewell. Causal inference in statistics: A primer. In *John Wiley & Sons*, 2016. 3, 11

[24] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference using invariant prediction: identification and confidence intervals, 2015. 2

[25] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *CoRR*, abs/1902.10811, 2019. 3

[26] 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, 2019. 2, 13

[27] Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *CoRR*, abs/1804.00792, 2018. 7

[28] Aman Sinha, Hongseok Namkoong, and John Duchi. Certifiable distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018. 6

[29] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xiansheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation, 2020. 11